



Renovació complerta del sistema d'informació intern
del Centre d'Educació Especial Jeroni de Moragas (IRENE)

TREBALL FINAL DE MÀSTER

Alumne: Carles Torras Font

Tutor a l'Empresa: Ana Maria Valls | Carolina Rofes

Tutor a la UOC: Víctor Robert Roca

Especialitat: Direcció de sistemes d'informació
en entorns de programari lliure

Móra d'Ebre, 17 de juny de 2015

Agraïments

Direcció d'AJEM i CEE per valorar i impulsar al projecte
Professionals del CEE per la seva participació i actitud
Universitat Oberta de Catalunya

I en especial:

Ana Valls i Carol Rofes

Víctor Robert

I a la família, que sempre hi és



Document amb llicència lliure de *Creative Commons*
Reconeixement – Compartir Igual (CC BY-SA)

Índex

Índex	i
Índex de figures	iii
Abreviatures	iv
Introducció	1
1.- Estudi de viabilitat	3
1.1.- Estat inicial del sistema	3
1.2.- Requisits plantejats, establiment de l'abast del sistema i del projecte	5
1.3.- Estudi i valoració d'alternatives. Selecció de la solució.	7
1.3.1.- A1. Establiment del programari de base necessari	7
1.3.2.- A2. Establiment del servei de gestió de la informació	10
1.3.3.- Riscos generals del projecte	12
2.- Anàlisi del sistema	13
2.1.- Definició del sistema, establiment de requisits i interfícies d'usuari	13
2.1.1.- Descripció de l'acció A1: Programari base	13
2.1.2.- Descripció de l'acció A2: Serveis de gestió de la informació	15
2.1.3.- Resum dels requeriments	18
2.2.- Especificació del pla de proves	19
2.2.1.- A1: Programari de base	19
2.2.2.- A2: Serveis de gestió de la informació	19
3.- Arquitectura	21
3.1.- Estat actual	21
3.1.1.- Especificacions actuals del maquinari d'ús comú d'AJEM	22
3.1.2.- Especificacions actuals de maquinari i programari del CEE	22
3.1.3.- Tasques del projecte i enfocament del disseny	25
3.2.- Definició de nivells d'arquitectura	26
3.2.1.- Programari de base (T3) i nova infraestructura	26
3.2.2.- Bloc Educatiu (T2.1)	27
3.2.3.- Sistema de Gestió de la Informació (T2.2)	30
3.3.- Especificació d'estàndards, normes de disseny i construcció	34
3.4.- Identificació dels subsistemes	34
4.- Casos d'ús	37
4.1.- Revisió de casos d'ús per subsistema	37
4.1.1.- Bloc educatiu (T2.1)	37
4.1.2.- Sistema de Gestió de la Informació (T2.2)	39
4.1.3.- Programari de base (T3)	43
4.2.- Especificacions de desenvolupament i proves	44
4.2.1.- Bloc Educatiu	45
4.2.2.- Sistema de Gestió de la Informació	46
4.2.3.- Programari de base (T3)	48
4.3.- Requisits d'implantació	49
5.- Desenvolupament	51
5.1.- Planificació de les activitats d'integració de sistema	51
5.2.- Elecció de llicències	53

5.2.1.- Programes desenvolupats: Bloc educatiu i Gestor de la Informació	53
5.2.2.- Nou programari de base instal·lat als ordinadors del CEE	54
5.2.3.- Guies d'ús desenvolupades i document de Política d'IT del CEE	54
5.2.4.- Documents que conformen el TFM i Pràctiques Externes (lliurables de la UOC)	55
5.3.- Entorn de desenvolupament	55
5.3.1.- Bloc educatiu	56
5.3.2.- Sistema de Gestió de la Informació	58
5.4.- Documentació	63
6.- Implantació	65
6.1.- Formació	66
6.2.- Implantació del sistema i proves	67
6.2.1.- Bloc Educatiu	67
6.2.2.- Sistema de Gestió de la Informació	68
6.2.3.- Nou programari de base	69
6.3.- Nivell de servei	70
6.3.1.- Bloc Educatiu	71
6.3.2.- Sistema de Gestió de la Informació	71
6.3.3.- Nou programari de base	72
6.4.- Acceptació del sistema	72
7.- Manteniment	75
Bibliografia	77
Annex 1: Cronogrames de les activitats d'integració de sistema	79
Annex 2. Seguiment de les fases desenvolupament i implantació	81
A2.1.- Actes de reunió	81
A2.2.- Informes de seguiment	101
Annex 3. Taules de les bases de dades	107
A3.1.- Taules de la base de dades del Bloc Educatiu	107
A3.2.- Taules de la base de dades del Sistema de Gestió de la Informació	108
Annex 4. Codi generat en la fase de desenvolupament	115
A4.1.- Bloc Educatiu	115
A4.2.- Sistema de Gestió de la Informació	144
A4.3.- Gestor de còpies de seguretat	265
Annex 5. Documentació generada	275
A5.1.- I001. Política d'IT del CEE	275
A5.2.- I002. Guia d'ús del Bloc Educatiu	279
A5.3.- I003. Guia d'ús del Gestor d'Informació	283
A5.4.- I004. Guia d'ús dels ordinadors del CEE	293

Índex de figures

Figura 1. Emplaçament dels departaments d'AJEM.....	3
Figura 2. Plànol de situació de les sis aules del CEE.....	4
Figura 3. Ús de SO distribucions de SO lliures al 2014.....	9
Figura 4. Infraestructura d'IT actual d'AJEM. Diagrama de components.	21
Figura 5. Plànol de situació de les aules del CEE	22
Figura 6. Infraestructura d'IT projectada d'AJEM. Diagrama de components.	26
Figura 7. Diagrama UML de components del Bloc Educatiu.	28
Figura 8. Diagrama UML de components amb interfícies del Bloc Educatiu.....	29
Figura 9. Diagrama de situació de targetes CRC pel Bloc Educatiu.....	30
Figura 10. Diagrama UML de components del Sistema de Gestió de la Informació.	31
Figura 11. Diagrama UML de components amb interfícies del Sistema de Gestió de la Informació.....	32
Figura 12. Diagrama de situació de targetes CRC pel Sistema de Gestió de la Informació.....	33
Figura 13. Diagrama de situació pel Bloc Educatiu	35
Figura 14. Diagrama de situació pel Sistema de Gestió de la Informació	35
Figura 15. Diagrama UML de revisió de casos d'ús amb components implicats pel Bloc Educatiu.....	37
Figura 16. Diagrama entitat/relació del Bloc Educatiu	38
Figura 17. Diagrama UML de classes (model relacional) del Bloc Educatiu.....	39
Figura 18. Diagrama UML de revisió de casos d'ús amb components implicats pel Sistema de Gestió de la Informació.....	40
Figura 19. Diagrama entitat/relació del Sistema de Gestió de la Informació	41
Figura 20. Diagrama UML de classes (model relacional) del Sistema de Gestió de la Informació.....	43
Figura 21. Esquema Model-Vista-Controlador del Sistema de Gestió de la Informació.....	47
Figura 22. Cronograma de la fase de desenvolupament.....	52
Figura 23. Imatge d'una interfície del Bloc Educatiu.....	56
Figura 24. Estructura del codi del Bloc Educatiu	57
Figura 25. Imatge d'una interfície del Sistema de Gestió de la Informació.....	59
Figura 26. Estructura del codi i projecte del Sistema de Gestió de la Informació.....	61
Figura 27. Interfície de l'aplicació de còpies de seguretat del Sistema de Gestió de la Informació	62
Figura 28. Cronograma de la fase d'implementació	65
Figura 29. Procés d'implantació del nou programari de base.....	69

Abreviatures

AJEM	Associació de Rehabilitació Jeroni de Moragas
BD	Base de Dades
CEE	Centre d'Educació Especial
CET	Centre Especial de Treball
GNU	GNU <i>is not</i> Unix
IT	Tecnologies de la Informació
LDAP	Protocol Lleuger d'Accés a Directoris
LOPD	Llei de Protecció de Dades
NTFS	En aquest context, <i>New Technology File System</i> , sistema d'arxius de Microsoft® Windows®
PDF	Portable Document Format
PL	Programari Lliure
PO	Paquet Ofimàtic
PrExt	Pràctiques Externes
RP	Resta de Programari
SGBD	Servei de Gestió de Base de Dades
SO	Sistema Operatiu
TFM	Treball Final de Màster
TO	Taller Ocupacional
WIFI	Xarxa sense fils (<i>fidelitat</i> sense fils)

Introducció

L'Associació de Rehabilitació Jeroni de Moragas (AJEM), localitzat a Móra d'Ebre (Tarragona) és una organització que gestiona un Centre d'Educació Especial (CEE), un Taller Ocupacional (TO), un Centre Especial de Treball (ET) i una residència per persones amb necessitats especials. Fou fundat fa trenta anys i dóna servei a les comarques de la Ribera d'Ebre, Priorat, Terra Alta, Baix Camp i Baix Ebre.

En general en tot AJEM i també específicament en el CEE, on s'emmarca aquest projecte, no ha existit mai una gestió específica dels serveis d'IT. A l'inici d'aquest projecte, disposaven de certa infraestructura informàtica, la qual s'ha anat implementant al llarg del temps sense cap política i cobrint necessitats puntuals (per exemple, cada ordinador disposa d'un programari de base diferent). En quan a serveis d'IT, es disposava d'un domini mitjançant el qual es té servei de correu electrònic i pàgina web, ambdós serveis externalitzats. Internament, només es disposa d'un servei de còpies de seguretat de fitxers. Tanmateix, no existia cap servei de gestió de la informació.

Així, mitjançant l'execució d'aquest TFM, s'ha pretès executar la primera fase d'un projecte més ampli, proposat pel propi autor. El projecte complert inclou diverses accions que resumidament són les següents. En primer lloc, concloure la implementació de la infraestructura d'IT (sobretot, manca cablejar tot el CEE). En segon lloc, definir una política de programari de base (definir quins programes calen a partir de les necessitats, incloent el propi sistema operatiu, i unificar-los en totes les màquines del centre) i gestió de la informació (definir els serveis necessaris). En tercer lloc, implementar el paquet complert de serveis al servidor d'AJEM. En quart lloc, implementar el programari de base establert en tots els ordinadors del CEE i migrar l'esquema actual de treball basat en ordinadors locals a ús dels ordinadors locals com a simples terminals els quals treballen contra servidor.

En aquesta primera fase corresponent al TFM, s'ha pretès crear la política d'IT, implementar el programari de base unificat a tots els ordinadors locals del CEE i crear els primers serveis de gestió de la informació. Com a accions destacables, es deixen per fases posteriors completar la infraestructura i migrar de sistema de treball basat en servidor. Per assolir el darrer, és imprescindible garantir el servei de xarxa intern.

Aquest document inclou totes les parts que componen un projecte de direcció de sistemes d'informació que són l'estudi de viabilitat, l'anàlisi i el disseny del sistema, el desenvolupament, implantació i manteniment, segons la metodologia descrita en els materials de l'assignatura del TFM [1] i alguna referència més genèrica [2].

1.- Estudi de viabilitat

Aquest capítol inclou els següents apartats:

- Estat inicial del sistema.
- Requisits plantejats, establiment de l'abast del sistema i del projecte.
- Estudi i valoració d'alternatives. Selecció de la solució.

En el primer punt es descriu quin és la situació actual al CEE i a l'AJEM en quan a IT. En el segon punt es comenta a quin situació es voldria arribar. Donat que les accions a emprendre són nombroses i per realitzar aquest projecte es tenen recursos limitats, s'identifica quin és l'abast del projecte. També s'estableix quin és l'abast del nou sistema a implementar. Finalment, es realitza un estudi i valoració d'alternatives per poder aconseguir arribar a la nova situació plantejada i es realitza una primera elecció de quin serà, amb tota probabilitat, la millor solució (manca per enllestir l'inventari exhaustiu de tot el maquinari del centre, previst per una de les primeres tasques en l'execució del projecte, que permetrà prendre la decisió final).

1.1.- Estat inicial del sistema

Aquest projecte s'emmarca en el Centre d'Educació Especial (CEE) Jeroni de Moragas, el qual forma part de l'Associació de Rehabilitació Jeroni de Moragas (AJEM). A part de residències que també gestiona, geogràficament, AJEM està concentrada en un sol emplaçament. En aquest, com mostra la figura 1, s'hi poden distingir tres zones: la corresponent al Taller Ocupacional i Centre Especial de Treball, la corresponent a Administració i la corresponent al CEE.



Figura 1. Emplaçament dels departaments d'AJEM

Ni AJEM en general ni el CEE particularment no han disposat mai d'una estratègia d'IT i el seu desplegament, als llarg dels anys, s'ha efectuat d'una manera desordenada i en funció dels recursos disponibles tot i cobrint necessitats puntuals.

Actualment, en quan a maquinari, el CEE disposa de vuit ordinadors, impressores i antenes receptores de WIFI repartits en 6 aules (figura 2)¹. Cada màquina disposa d'un programari de base diferent, en funció, sobretot, del seu origen. En relació al sistema operatiu, totes les màquines el tenen de la família Microsoft® Windows® [3], per bé que diverses versions. En quan a serveis, els usuaris del CEE disposen del servei de carpeta en xarxa sobre la qual es fan còpies de seguretat i tots els ordinadors disposen de connexió a Internet, irregular i deficient. Això és així per dos motius: perquè el senyal els arriba via WIFI amb cobertura limitada (en cap cas per cable) i perquè l'ample de banda que AJEM té contractada és de 10 MB. I d'aquesta connexió se n'alimenta no només els ordinadors del CEE, sinó els portàtils dels alumnes, tots els ordinadors del departament d'administració, tots els ordinadors del Taller Ocupacional i Centre Especial de Treball i tots els telèfons mòbils de totes les persones que en tenen (professionals i usuaris).

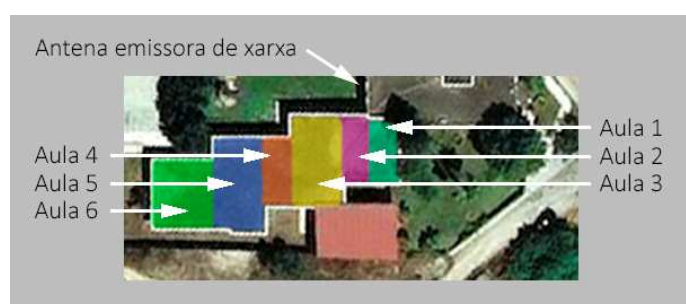


Figura 2. Plànol de situació de les sis aules del CEE

Cada professional del CEE treballa en la màquina que té a la seva aula, la qual i com s'ha comentat, té el seu propi programari. Així, cada màquina funciona de manera independent (es treballa 100% en local) i a com a servidor propi de cada usuari. Cada professional desa les seves dades en arxius (per exemple la llista d'alumnes, informes, reunions, etc.) en el seu ordinador local i si vol disposar de còpies de seguretat, cal que manualment desi també els arxius a la carpeta d'un servidor que està en xarxa. Tota la informació es desa en arxius i no hi ha cap tipus de gestió de la informació.

En particular i en relació a arxius, els professionals del CEE realitzen un número important de fotografies durant el curs escolar en relació a les activitats que desenvolupen. Aquestes fotografies les emmagatzemen en els disc durs locals dels ordinadors de les seves respectives classes, sense seleccionar-les (les guarden totes) i sense modificar-ne la mida. Eventualment, algunes d'aquestes fotografies també les emmagatzemen a la carpeta de xarxa sobre la qual es realitzen les còpies de seguretat, fet que compromet el servei en quan a capacitat i velocitat de còpia.

A banda dels recursos informàtics propis del CEE, aquest també es beneficia de la infraestructura comuna d'AJEM. A banda de la connexió a Internet ja mencionada (i del corresponent encaminador² propietat del proveïdor de serveis d'internet -ISP-), la resta d'infraestructura informàtica de la qual disposa AJEM és la següent (maquinari ubicat a la segona planta de l'edifici d'administració, figura 1):

¹ Cal fer un inventari complet d'aquest maquinari on s'inclouï no només l'especificitat dels ordinadors, sinó de tots els components i controladors. Aquesta feina és de cert volum i conseqüentment, queda fora de la fase de planificació i forma part d'una de les primeres tasques de la d'execució.

² <http://www.movistar.es/particulares/atencion-cliente/internet/adsl/equipamiento-adsl/routers/observa-aw4062/>

- Màquina servidor. Correspon a un model *HP Proliant ML110 G7*³ amb 2 Gb de memòria RAM i dos discs durs de 250 Gb. Disposa d'un sistema operatiu Windows Server 2008 R2. Aquest ordinador no està connectat a Internet i només dóna servei a nivell d'Intranet. L'únic servei que proporciona és el de còpies de seguretat d'arxius, mencionat anteriorment. Per això, usa el programa *Cobian Backup II*⁴. La metodologia implementada (a través d'un proveïdor extern) és la següent: els usuaris desen arxius en una partició d'un dels dos discs durs del servidor (el servidor té 2 discs durs, un té dues particions: una pel SO i programes i l'altre per dades). En temps real, el programa de còpia sincronitza els fitxers en l'altre disc dur del servidor. I una vegada al dia (a les 24h), el programa de còpia sincronitza les dades a un tercer disc dur USB extern (*backup* incremental). Finalment i addicionalment, es disposen de 2 discs durs USB externs els quals es van intercanviant un cop a la setmana.
- Disc durs externs. Es disposa de 2 discs durs externs USB d'1 TB cadascun, els quals s'utilitzen per realitzar còpies de seguretat, tal i com s'ha descrit en l'anterior punt.
- Un sistema d'alimentació ininterrompuda (SAI). Correspon a un model de la marca *Netway Technology*.
- Commutador (switch). Es tracta d'un model *OvisLink EVO FSH24*⁵. Distribueix xarxa al Taller Ocupacional i Centre Especial de Treball per cable i a l'Escola (CEE) via una antena amplificadora (i els ordinadors del CEE disposen de dispositius receptors del senyal sense fils).

En relació als serveis de pàgines web i correu electrònic, aquests estan externalitzats. El CEE no disposa de pàgina web pròpia, sinó que està integrada dins de la d'AJEM. AJEM té adquirit el domini *ajem.cat* sota el qual existeix el lloc web. Del disseny del mateix, manteniment i allotjament, se n'ocupa l'empresa externa NUWA Agència Digital [4], la qual, alhora, usa el servei d'una tercera empresa: CDMon [5]. En relació al servei de correu electrònic, els professionals del CEE usen el que els subministra XTEC (Departament d'Ensenyament, Generalitat de Catalunya) i el gestionen via web (interfície d'XTEC). Els professionals del Taller Ocupacional, Centre Especial de Treball i Administració d'AJEM utilitzen adreces de correu electrònic del domini propi *ajem.cat*, però la gestió està també externalitzada. En conclusió, aquests serveis ja estant resolts, no es precisa cap canvi i per tant, queden ja d'entrada fora de l'abast del projecte. Així, es constata que l'abast d'aquest projecte té sobretot incidència interna al CEE.

Finalment, cal esmentar que ni el CEE ni AJEM en general disposa de cap persona específica que s'ocupi tècnicament d'affers d'IT. És el gerent d'AJEM qui ho gestiona i realitza simples accions puntuals.

1.2.- Requisits plantejats, establiment de l'abast del sistema i del projecte

Donada la situació actual descrita en l'anterior apartat, en relació a la IT del CEE es detecten les següents importants i principals mancances:

- Diversitat de programari en cada màquina i manca de control sobre el mateix.
- Manca de gestió d'IT.
- Manca total de base de dades i gestió de les mateixes. Tot funciona a través únicament d'arxius. Això inclou també el control/comptes d'usuaris.

³ http://h20564.www2.hp.com/hpsc/doc/public/display?docId=emr_na-c02793565

⁴ <http://www.cobiansoft.com/cobianbackup.htm>

⁵ http://www.ovislink-espana.com/es/4_4int_1_18/Productos/Switches/Fast-ethernet/Evo-FSH24/

- Mancança en infraestructura. Per exemple, cablejat de xarxa al CEE.

Aquestes mancances identificades, doncs, fan palesa la necessitat de dissenyar i executar una estratègia de centre en quan a IT que hi posi remei. La possible execució d'un projecte complet hauria d'assolir els següents objectius tècnics:

- **Completar la infraestructura de xarxa.** Aconseguir que totes les màquines del centre estiguin connectades a xarxa via cable. Reservar el WIFI per altres ordinadors d'usuaris, telèfons mòbils, etc.
 - Augmentar la capacitat contractada d'ample de banda del servei d'Internet.
- **Establiment d'una política de centre** en quan a programari de base. Identificar tot el programari necessari pel centre (paquet ofimàtic, tractament d'imatges, tractament de so, etc.), elegir-lo unificadament i implementar-lo.
- **Canvi d'orientació de treball cap a ordinadors terminals.** Actualment, el mètode de treball és gairebé cent per cent local. És a dir, cada màquina del CEE té el seu control d'usuaris, de gestió del maquinari (totes les impressores es connecten via USB a l'ordinador corresponent), de gestió de fitxers, etc. En quan a la gestió de fitxers, en general, els usuaris treballen en carpetes locals i eventualment, copien arxius en la carpeta en xarxa, sobre la qual es realitzen les còpies de seguretat. Això dificulta molt el manteniment de cada màquina, no només pel programari particular que té, sinó pel maquinari que té instal·lat, les dades que conté, etc. Així, cal arribar a una situació en la que es treballi al màxim contra servidor (gestió d'usuaris, carpetes d'usuari de xarxa) i que el maquinari es connectés en xarxa. En definitiva, que els ordinadors fossin simples clients. Això, no només tindria l'avantatge de facilitar molt el manteniment de cada màquina, sinó que tots els usuaris podrien utilitzar qualsevol ordinador, fins i tot el de casa si es permetés tenint en compte altres aspecte no tècnics, com per exemple legals (LOPD).
- **Gestió de la informació.** Actualment, només es treballa en fitxers. Tota la informació està desada en els mateixos (típicament, de processador de texts). Cal instaurar un sistema de treball basat en dades. Així, és necessari posar en marxa una base de dades, un gestor de base de dades i una interfície web de connexió per l'usuari que li permeti la gestió de les mateixes. S'hi ha de gestionar: aules, alumnes, patologies, medicacions, incidències, reunions, etc.

Tanmateix, l'abast d'aquest Treball Final de Màster no pot integrar tot el projecte complet per manca de temps, pressupost, etc. Així, es proposa dividir el projecte sencer en diverses fases. La primera fase correspondria a l'abast del TFM i inclouria les següents accions, adequades pels recursos disponibles:

- A1. Establiment del programari de base necessari. Elecció del mateix (basat en Programari Lliure tant com sigui possible) i implementació.
- A2. Establiment de serveis de gestió de la informació. Posada en marxa de la BD i SGBD. Programació de la interfície d'usuari adaptada a les necessitats del CEE i implementació.

La consecució de les anteriors accions, conduirà a l'establiment de la política d'IT del CEE, el qual quedarà reflectit en un document oficial.

Així, quedarà per una segona o més fases completar la infraestructura del CEE i migració d'ús dels ordinadors client cap a ordinadors terminals. Aquest darrer aspecte s'aconseguirà implementant nous serveis al servidor central del CEE, com per exemple, el servei de control d'usuaris (servei tipus LDAP). Tanmateix, per aconseguir això darrer cal disposar d'una xarxa fiable i això no serà així fins que la tasca

de completar la infraestructura estigui complerta. Això últim també implica efectuar certa despesa, fet que en aquests moments no és viable.

1.3.- Estudi i valoració d'alternatives. Selecció de la solució.

L'estudi i valoració d'alternatives es realitza sobre les dues accions identificades en l'apartat anterior: l'establiment del programari necessari (A1) i l'establiment de servei de gestió de la informació (A2).

1.3.1.- A1. Establiment del programari de base necessari

Una de les tasques del projecte és establir conjuntament amb els professionals del CEE quin és el programari de base necessari. En molt bona mesura, això ja s'ha realitzat en l'etapa de planificació. Donat també que no es fa servir cap programa específic en el CEE (sí en el departament d'administració, que no està en l'abast del projecte, com el programa de gestió de nòmines) es pot fer una divisió en tres grans grups: sistema operatiu (SO), paquet ofimàtic (PO) i resta de programari (RP). Dins de la resta de programari, segurament, s'inclou un programa de tractament d'imatges, un de tractament de so, un gravador de CD/DVD i un reproductor multimèdia amb paquet de còdecs complerts.

Com s'ha descrit, en l'actualitat, cada màquina disposa d'un programari divers. Algunes coincidències, tanmateix, són que totes elles disposen d'un SO de la família Microsoft® Windows® encara que amb versions diferents i del PO Microsoft® Office® i també amb versions diferents. A partir d'aquí, la resta de programari és del tot variat. Malgrat que en aquests moments l'inventari no és exhaustiu, s'estima que aproximadament la meitat de llicències de SO no estant regularitzades i que un 75% de les del paquet ofimàtic tampoc (són versions de prova, etc.). Això correspon a 4 i 6 llicències respectivament.

Així, s'entreveuen tres grans alternatives:

1. Mantenir Microsoft® Windows® com a SO [3], mantenir Microsoft® Office® [6] com a PO i disposar de PL per tota la resta.
2. Mantenir Microsoft® Windows® com a SO, migrar a LibreOffice [7] o OpenOffice [8] com a PO lliure i disposar de PL per tota la resta. LibreOffice serà l'elegit en aquest cas, donat l'estat actual dels dos projectes (OpenOffice està en procés de canvi i LibreOffice, actualment, gaudeix de millor desenvolupament i és ara el líder⁶).
3. Migrar a un SO lliure, migrar a LibreOffice com a PO lliure i disposar de PL per tota la resta.
 - a. SO lliure a implementar. Existeix una àmplia varietat de distribucions [9]. A sota, es realitza una discussió al respecte.

Una opció que ja es descarta des del principi és que la RP no estigui basat en PL ja que no hi ha cap motiu perquè sigui així. Ni s'està habituat en general en cap altre programa, ni hi ha llicències disponibles d'un programa privatiu i no gratuït, etc.

⁶ <http://www.datamation.com/open-source/libreoffice-vs.-openoffice-why-libreoffice-wins-1.html>

Valoració econòmica

Partint de la base de càlcul que el preu mig de llicència de Microsoft® Windows® Pro versió educativa és de 60,00 euros i que la de Microsoft® Office® Home versió educativa és de 119,00 euros, la inversió per cadascuna de les opcions és de 954,00 euros, 240,00 euros i 0,00 euros respectivament.

Aquesta valoració econòmica, intrínsecament, garanteix la viabilitat legal del programari, essent que, addicionalment, no s'entreu cap condicionant que pugui implicar deixar de complir els requeriments dels diversos tipus de llicències. Per exemple, si s'utilitza un programa amb llicència GNU/GPL, no es preveu que ningú pugui usar el codi, modificar-lo i convertir-lo en propietari.

Valoració tècnica

De les tres categories, la que té més risc de viabilitat és la corresponent al SO. Normalment per dos motius: pels programes que s'hi pot instal·lar i per la disposició de controladors dels diversos components. En aquest cas, no hi ha risc pel primer punt ja que no hi ha cap requeriment de programa especial. Sí que n'hi pot haver pel segon. Però en aquest cas també és mínim. Els components principals externs que existeixen són impressores i receptors WIFI. A manca de realitzar l'inventari específic i corroborar-ho, es sap que tots els productes són prou estàndards i habituals com perquè no hi hagi manca de controladors. La mateixa reflexió es pot realitzar sobre els components interns. A més, per l'ús que es dona a les màquines, controladors estàndards serien suficients.

La viabilitat tècnica de les altres categories sembla garantida o en tot cas, no s'entreu cap diferència significativa de risc tècnic entre els diversos programaris.

Valoració operativa

Com s'ha comentat, els usuaris estant habituats a usar Microsoft® Windows® i Microsoft® Office®. Per tant, existeix un risc al canviar aquests programes.

En relació al SO, l'ús que els usuaris en fan es molt bàsic. És a dir, no usen funcions avançades del sistema, ni en general realitzen funcions de configuració, etc. És a dir, bàsicament el posen en funcionament i el paren, i n'executen el programari instal·lat. Per tant, si es substitueix per un SO que permeti fer aquestes funcions bàsiques d'una manera molt similar i que el canvi vagi acompanyat de formació i de guies d'usuari, el risc pot quedar ben minimitzat.

Valoració de distribucions de SO lliure

Un aspecte que conflueix entre la valoració tècnica i social és l'elecció de la distribució de SO lliure a elegir, si aquesta és finalment l'opció escollida. Hi ha centenars de distribucions. Un anàlisi exhaustiu de les mateixes podria ser objecte d'una tesi. En aquest cas, cal simplificar-ne l'anàlisi. Un primer criteri a aplicar és triar-ne una d'entre les més usades actualment. Encara que és difícil predir quin serà el futur de totes elles, les més populars actualment corresponent a les que tenen més gent involucrada (tant desenvolupadors com usuaris) i per tant, les que més poden anar progressant i de les que més suport es pot aconseguir.

La figura 3 mostra un exemple d'estadístiques d'ús de distribucions de SO lliures (versió escriptori, no servidor) a 2014. Ubuntu [10] i Arch Linux són les clares capdavanteres. Tanmateix, aquestes dades també s'han de contrastar amb dades evolutives anuals i tradició de les mateixes. En aquest darrer sentit, hi ha cert consens que Ubuntu, OpenSUSE i Fedora són les tradicionalment més implantades. Així, Ubuntu és clarament una de les capdavanteres, tant per història com per ús actual. Per tant, Ubuntu és una distribució candidata de partida.

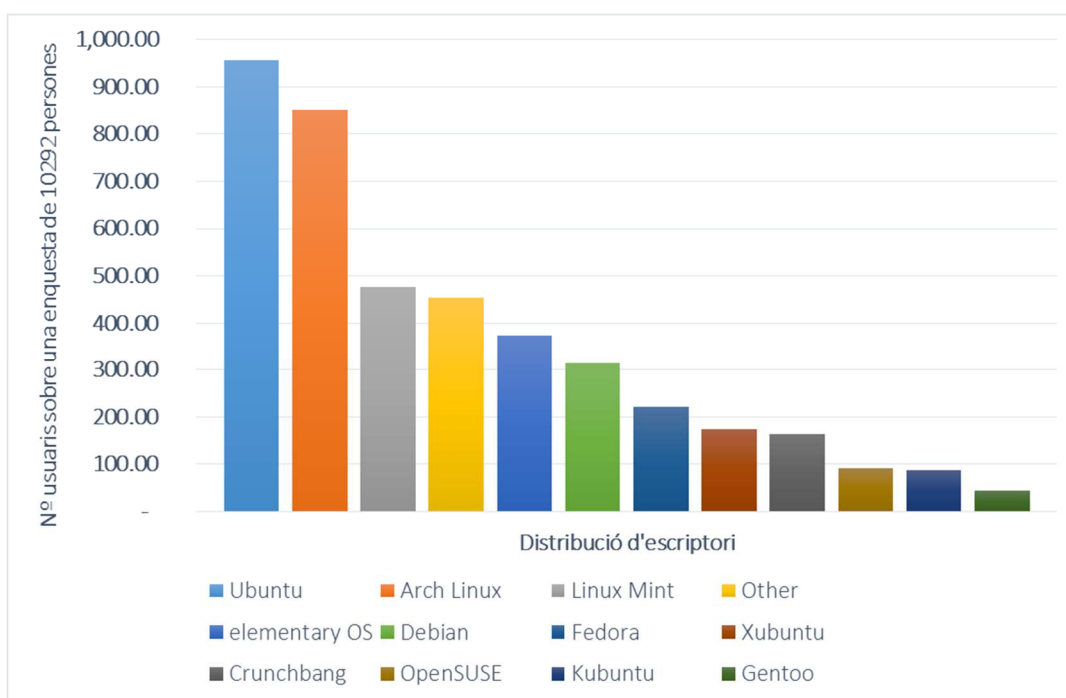


Figura 3. Ús de SO distribucions de SO lliures al 2014.

(font: <http://brashear.me/blog/2014/05/18/results-of-the-2014-slash-r-slash-linux-distribution-survey/>)

Tanmateix, de la valoració social efectuada, se'n desprèn la clara necessitat que el sistema elegit sigui tant proper al Microsoft® Windows® (a nivell d'ús d'usuari bàsic), com sigui possible. I és en aquest sentit que Linux Mint [11] es pot convertir en l'opció final a elegir pels següents motius. D'entrada, està basada en Ubuntu i també usa el seu repositori (per tant, basada en la distribució líder). En segon lloc, és la distribució que més s'aproxima a Microsoft® Windows®. En tercer lloc, és una distribució en clar ascens. Segons *distrowatch* [12] es manté com la més popular en els últims mesos (per la xarxa es poden trobar molts més anàlisis [13-15]). Finalment, una darrera avantatge és que la distribució Linux Mint ja porta incorporada per defecte altre programari a instal·lar com LibreOffice, Firefox, GIMP, VLC, etc. I això suposarà una facilitat durant l'etapa d'implementació.

Elecció de la solució

Donat que la part econòmica és una de les més importants i que en les valoracions tècniques i econòmiques no hi ha especial risc ni preferència en cap de les 3 opcions, l'opció que s'entreu com a més adequada és la tercera (migrar a un SO lliure, migrar a LibreOffice com a PO lliure i disposar de PL per tota la resta).

Tanmateix, en aquests moments, la decisió queda encara en condicional ja que manca confirmar alguns aspectes tècnics, com s'ha explicat. No es preveu, però, que sorgeixin inconvenients. Si sorgissin, afectarien sobretot només al SO, per la qual cosa, en situació extrema, s'optaria per la segona opció (mantenir Microsoft® Windows® com a SO, migrar a LibreOffice com a PO lliure i disposar de PL per tota la resta), quedant la primera descartada.

1.3.2.- A2. Establiment del servei de gestió de la informació

Aquesta acció inclou dos àmbits d'escenaris a avaluar: el programari a usar i/o desenvolupar i sobre quina infraestructura instal·lar-ho.

En quan al programari, el concepte fonamental és dissenyar una eina de gestió dels recursos bàsics del CEE i les seves aules. Donada la dimensió del CEE, la quantitat d'informació és reduïda i la quantitat de recursos són reduïts (aules, alumnes, mestres, eines, etc.). Tenint en compte això, es poden plantejar dues grans alternatives:

1. Ús de programari existent i adaptació del mateix.
2. Desenvolupar programari de nou.

Valoració econòmica. Sense haver entrat en molta profunditat, existeix un cert nombre de solucions lliures per aquest cas, implementades en diversos centres educatius. Exemples són el projecte *Sakai* [16] o el *Moodle* [17]. Així, cap de les dues opcions podria suposar un cost d'adquisició de llicència. I donat que la mà d'obra per implementar-les va a càrrec de l'estudiant, no s'entreveu cap més cost.

Valoració tècnica. Ambdues solucions poden ser viables tècnicament. Tanmateix, la primera opció presenta dos inconvenients. D'entrada, els requisits d'implementació (requeriments de programari i maquinari) poden ser més elevats. Per l'abast del projecte i per la segona opció, un simple servidor web podria ser suficient. I en segon lloc, l'abast de les solucions existents són molt més àmplies que el que es projecte en aquest cas. Per tant, s'entraria en un nivell de complexitat de gestió i funcionament del programa innecessari per aquest cas.

Valoració operativa

Aquesta valoració és de la més important en aquest cas. Les solucions existents poden tenir l'inconvenient d'estar més orientades als alumnes que als professors, o almenys, en igualtat de condicions i contemplant molt la interacció entre els dos rols. Però en aquest cas, el servei va destinat, pràcticament, només al professor. És a dir, aquest cas es podria definir com un campus virtual orientat al professor, a camí d'un sistema de gestió interna d'una empresa.

Un segon punt és que el CEE s'allunya d'una situació estàndard de centre educatiu com pot ser una escola convencional, un institut o una universitat. Es pot fer el paral·lelisme amb la disponibilitat de material editorial. Mentre que les escoles convencionals disposen d'una àmplia oferta de llibres i editorials, això és inexistent pel CEE donat que les classes no són gens homogènies en quan a alumnes, els quals tenen nivells i patologies molt diferents entre ells. Per tant, no hi ha un mateix llibre que pugui servir per tota la classe. I en aquest mateix sentit, la personalització necessària del programa pot no trobar solució en programari disponible que està enfocat a solucions més estàndards.

Un tercer punt, és la facilitat d'ús de l'eina per part del professional. El professional requereix la màxima facilitat d'ús. I una eina feta a mida per gestionar la poca informació que cal ha de ser molt més simple d'usar que una eina com *Moodle* o *Sakai* que té certa complexitat.

I, finalment, un quart punt és la facilitat de manteniment del servei. Es requereix que el sistema es mantingui de la forma més autònoma possible. Fer l'eina a mida, amb la funcionalitat justa i necessària, pot no només minimitzar molt el manteniment sinó la possibilitat de fallades que requereixin intervenció.

En aquest cas, no s'entreveuen diferències rellevants a nivell de *valoració legal*.

Elecció de la solució

En aquest cas, totes les valoracions apunten a l'elecció de programar una eina a mida, la qual s'ha de basar en la màxima senzillesa aprofitant la limitada funcionalitat que ha de tenir i baix volum de dades que ha de gestionar. Fet que no treu la utilitat que pugui tenir.

En quan a la infraestructura a utilitzar per implementar el servei, s'entreveuen dues possibilitats:

1. Ús d'infraestructura interna.
2. Ús d'infraestructura externa.

Valoració econòmica. Després de realitzar un primer inventari al CEE sobre maquinari disponible (descriu a l'anterior secció d'aquest document), s'ha conclòs que existeixen tots els recursos necessaris i per tant, no seria necessària cap inversió, cost de lloguer o servei, etc., per l'opció 1. En quan a l'opció 2, s'ha trobat un servei extern el qual posaria en disposició del CEE els recursos necessaris per desplegar el servei sense cost econòmic. Així, el cost per les dues opcions és igual i zero.

Valoració tècnica. Per l'opció 2, no s'entreveu cap inconvenient ni risc en la seva aplicació. En canvi, per l'opció 1, sí que hi ha cert risc. Per realitzar la implementació, cal instal·lar en el servidor existent cert programari i realitzar algunes configuracions. Això comporta certs riscos que poden comprometre el funcionament del servei actual de còpies de seguretat i del propi sistema (no les dades en sí, de les quals es disposa de diverses còpies). Gestionant correctament les operacions, el risc pot ser minimitzat. Les condicions en que la infraestructura està en els dos casos és molt semblant (per exemple, el servei extern no està en un Centre de Dades, sinó en un espai semblant on està el maquinari central d'AJEM), per tant, no existeixen diferències en aquest sentit.

Valoració operativa. No s'entreveuen diferències significatives entre les dues opcions. Ja que totes les tasques es poden executar en xarxa i per tant, amb independència del lloc físic on estigui situat el maquinari.

Valoració legal. Aquest és un aspecte molt rellevant en aquest cas degut a la Llei de Protecció de Dades. El tipus de dades que es gestionen (personals, mèdiques, legals, etc.) fan que s'estigui en requeriments màxims de seguretat de gestió de les mateixes. Implementar el servei dins d'AJEM en servei exclusiu d'intranet, elimina riscos. Implementar-ho externament i per tant, amb necessitat de connexió a Internet, augmenta els riscos. Per aquest aspecte, AJEM compta amb un assessor especialitzat.

Elecció de la solució

Donat que en aquest cas, el requeriment i risc legal és clarament el de més importància, s'elegeix l'opció 1 (usar la infraestructura interna). Aquesta no comporta cap cost addicional (és disposa de tota la infraestructura) i només caldrà especial atenció en el risc de comprometre el servei actualment existent (es descriu en el capítol 2: anàlisi del sistema).

1.3.3.- Riscos generals del projecte

En aquest apartat, es llisten els riscos generals del projecte, el respectiu grau de rellevància previst, una breu descripció i mesura de contingència.

- Limitació de maquinari disponible. **Risc: 2**. *Contingència*: eventual disposició d'un mínim capital per inversió | ús de serveis externs gratuïts.
- Manca d'establiment de criteris per part dels professionals del CEE. **Risc: 3**. *Contingència*: reducció de l'abast del projecte.
- Resistència al canvi per part dels professionals del CEE. **Risc: 4**. *Contingència*: més esforços en formació, informació i generació de manuals.
- Excés d'accions a realitzar en relació a recursos disponibles. **Risc: 5**. *Contingència*: reducció de l'abast del projecte | ampliació del termini d'execució (fora del TFM oficial).
- Imprevistos per manca d'experiència o desconeixement tècnic de l'autor. **Risc: 2,5**. *Contingència*: suport del consultor i equip de la UOC.
- Incompliment de la Llei de Protecció de Dades. **Risc: 2**. El risc es minimitza inicialment en definir tots els serveis només en termes d'intranet i en comptar amb l'assessorament d'un expert. *Contingència*: incrementar aspectes de seguretat en la implementació | reduir dades de l'abast.
- Malmetre els serveis existents al CEE. **Risc: 2,5**. *Contingència*: Preveure còpies de seguretat | recórrer a qui va realitzar-ne la instal·lació.

Un aspecte també a considerar i que genera un risc és el manteniment posterior dels nous serveis implementats. Havent definit quines accions es duran a terme, s'entreveu un risc baix en aquest aspecte al concloure que no s'augmentarà significativament la necessitat de manteniment respecte la situació actual. És a dir, no es preveu que no es requerirà eventuais manteniment, però que o seran menors o iguals que els actuals.

- Es preveu que el nou programari de base sigui més robust i li calgui menys manteniment.
- Els serveis s'obtidran via web (intranet). Per tant, sense afectació als ordinadors locals.
- Els serveis que es prestaran via el servidor actual d'AJEM, usaran programari que a priori no necessita manteniment i la còpia de seguretat de dades s'integrarà dins l'actual sistema de còpies.

Hi ha dues contingències previstes: (1) usar els mateixos mecanismes que s'usen actualment davant de les necessitats de manteniment (qui el realitza actualment hauria de ser capaç de realitzar-lo en la nova situació) i (2) disposició de l'autor en continuar donant suport eventual.

2.- Anàlisi del sistema

D'acord amb el plantejament establert en la secció 1.2 d'aquest document, el present projecte inclou dues activitats:

- A1. Establiment del programari de base necessari. Elecció del mateix (basat en Programari Lliure tant com sigui possible) i implementació.
- A2. Establiment de serveis de gestió de la informació. Posada en marxa de la BD i SGBD. Programació de la interfície d'usuari adaptada a les necessitats del CEE i implementació.

A continuació, es descriuen agrupadament les especificacions de les accions anteriors, l'establiment dels requisits i la definició bàsica de la interfície d'usuari. Finalment, s'especifica el pla de proves.

2.1.- Definició del sistema, establiment de requisits i interfícies d'usuari

2.1.1.- Descripció de l'acció A1: Programari base

Aquesta acció requereix sobretot esforços en l'etapa de disseny i implementació, i menys en la de desenvolupament, ja que es preveu utilitzar el Programari Lliure tal qual. És a dir, no es preveu la necessitat d'adaptació o personalització de cap codi font.

L'etapa de disseny inclou dues tasques diferenciades d'entrada. La primera consisteix en identificar tots els tipus de programes necessaris a partir de les necessitats. Això s'ha realitzat conjuntament entre els professionals del CEE i l'autor: els primers han identificat les necessitats i a partir d'aquí, el segon ha proposat el programari que pot cobrir aquestes necessitats. La segona consisteix en realitzar un inventari del maquinari i programari actuals per diversos motius:

- Conèixer amb detall el programari instal·lat actual, les seves llicències, etc. No hi ha, actualment, cap inventari al respecte.
- Conèixer amb detall els controladors necessaris per avaluar les possibilitats d'instal·lació de sistemes operatius.
- Conèixer les dades emmagatzemades en cada ordinador les quals s'hauran de guardar.

L'etapa d'implementació també inclou dues tasques diferenciades: la instal·lació del nou programari de base i la formació als professionals i generació de manuals d'usuari. Com que la primera etapa s'anirà completant progressivament en els diversos ordinadors del CEE i al llarg d'algunes setmanes, pot ser precís realitzar, almenys, una primera formació abans de començar les instal·lacions.

Durant la present etapa de planificació, s'han efectuat algunes reunions amb els professionals del CEE les quals han permès identificar quines són les necessitats específiques en quan a programes.

En quan a SO, no hi ha una determinant necessitat específica per part dels usuaris més que unificar-lo en totes les màquines i no tenir dificultats importants en usar-lo (això implica que tingui una usabilitat semblant al sistema que utilitzen actualment). Pels usuaris, el SO només serveix de plataforma per executar la resta de programari. Tanmateix, en aquest cas, sí que existeixen altres necessitats més alienes a l'usuari, com són respectar el marc legal i minimitzar-ne el cost. Considerant tot això, el SO

que més pot cobrir les necessitats és un SO GNU/Linux (sense cost de llicències i de fàcil compliment de la seva llicència) que tingui una usabilitat el més semblant possible als sistemes actual basats en Microsoft® Windows® (Linux Mint).

En quan a PO, els usuaris plantegen certes necessitats que actualment no tenen cobertes (addicionalment a les que sí tenen). Aquestes són poder realitzar formularis protegits, inserir multimèdia a les presentacions i obrir documents PDF en l'editor de textos per poder-los editar. Analitzant-les, hom pot concloure que no és que el programari que utilitzaven fins ara no oferís aquestes funcionalitats, sinó que els usuaris no les sabien utilitzar. Tanmateix, per aquest punt, també hi ha les mateixes necessitats que pel SO, més alienes als usuaris, però que s'han d'assolir: respectar el marc legal i minimitzar el cost. LibreOffice (a diferència del programa usat actualment), cobreix totes aquestes necessitats a cost zero de llicència. Tanmateix, no només amb la seva implementació queden cobertes les necessitats, sinó que també es fa imprescindible formació als usuaris per que aprenguin com realitzar les tasques sol·licitades amb LibreOffice.

Finalment, en quan a la RT, sí que actualment no tenen cobertes les necessitats per manca de programari. Els usuaris tenen tres necessitats bàsiques: editar imatges, editar àudios i poder reproduir tot tipus d'arxius multimèdia. Alhora, tanmateix, també s'han de cobrir les necessitats de respectar el marc legal i minimitzar el cost de llicència. Així, es proposa implementar nou programari de base, basat en PL, que els cobreixi aquestes necessitats. Donat que són programes nous, també caldrà formació als usuaris perquè els sàpiguen fer funcionar (per l'editor d'imatges i so).

Es constaten també dos factors: que els usuaris no són experts en informàtica i que alhora, no els cal un coneixement profund de la funcionalitat dels diversos programes. La conseqüència del primer factor és posar èmfasi en la formació i generació de manuals bàsics. No es pot assumir que els usuaris, per ells mateixos i sense cap assistència, s'adaptaran adequadament al nou programari. Del segon factor es desprèn que no és necessari (de fet, seria contraproduent tenint en compte també el primer) un excés de formació o manuals complexes sobre totes les possibilitats que ofereix el programari. És a dir, es precisa formació i informació concisa, breu i clara, molt orientada només a les necessitats que tenen.

A partir de l'anterior anàlisi de necessitats i el de viabilitat, es proposa específicament la implementació del següent programari de base:

- Sistema operatiu. Segons l'estudi de viabilitat, l'opció elegida serà probablement Linux Mint [11]. La segona opció seria Microsoft® Windows® [3]. El SO no té cap requeriment no estàndard o no habitual, més enllà de contenir la resta de programari, suportar les impressores i els receptors WIFI existents, i suportar els serveis de xarxa (carpetes compartides en xarxa, client web). Els professionals demanden eines per poder mantenir el sistema net i protegit. Migrar a un SO GNU/Linux contribuirà ja per si mateix a aquesta millora.
 - Donat que el SO del servidor treballa amb Microsoft® Windows® Server 2008 R2, caldrà segurament dotar Linux Mint del protocol SAMBA que no porta per defecte. Això es pot assolir amb els paquets *samba*, *system-config-samba*, *cifs-utils* i *winbind* existents al repositori oficial. També pot ser necessària una eina d'auto-muntar particions NTFS de Microsoft® Windows® (paquet *ntfs-config*).

- Paquet ofimàtic. Segons l'estudi de viabilitat, l'opció elegida serà LibreOffice [7] (solució PL). Els professionals del CEE no precisen cap característica molt especial a banda de les usuals que s'inclouen en el paquet per defecte, per bé que sí que fan esments especial a tres característiques:
 - Disseny de formularis en el processador de texts, que es pugui protegir. LibreOffice ja incorpora aquesta funcionalitat per defecte. Caldrà donar formació específica als usuaris sobre la mateixa.
 - Obrir PDF per poder-los editar. Inicialment, això es podia realitzar amb LibreOffice via la instal·lació d'una extensió. En les darreres versions, sembla que és una funcionalitat ja base⁷. En qualsevol cas és possible. Caldrà donar formació específica als usuaris sobre la mateixa.
 - Incloure multimèdia a presentacions. LibreOffice Impress ja incorpora aquesta funcionalitat per defecte⁸. Caldrà donar formació específica als usuaris sobre la mateixa.
- Paquet de tractament d'imatges. Gimp [18] (solució PL) cobreix les necessitats bàsiques que tenen els usuaris. Caldrà donar formació bàsica als usuaris.
- Paquet de tractament de so. Audacity [19] (solució PL) cobreix les necessitats bàsiques que tenen els usuaris. Caldrà donar formació bàsica als usuaris.
- Reproductor i còdecs multimèdia. Un reproductor i còdecs ja estan inclosos en el propi sistema operatiu Linux Mint. Tanmateix, poden ser insuficients. Cal inventariar amb detall la necessitat i si fos el cas, les següents aplicacions (les quals incorporen còdecs) poder ser instal·lades⁹: MPlayer i SMPlayer, VLC, FFmpeg i GStreamer o ubuntu-restricted-extras. Cal considerar que no tots els còdecs corresponen a PL (però sí gratuït).

Finalment, caldrà desenvolupar un document escrit que constitueixi un llibre blanc de política d'IT del centre i un manual breu i simple d'usuari i preveure sessions de formació bàsica als professionals del CEE.

2.1.2.- Descripció de l'acció A2: Serveis de gestió de la informació

Aquesta acció inclou tasques en les tres fases principals d'execució d'un projecte: disseny, desenvolupament i implementació.

L'etapa de disseny consisteix principalment en una tasca, destinada a identificar aquells serveis de dades a implementar. S'ha realitzat conjuntament amb els professionals del centre, els quals han identificat quines dades són més necessàries de gestionar. S'ha definit el volum de dades tenint en compte els recursos disponibles, sobretot, de temps de programació.

L'etapa de desenvolupament es centrarà en dues grans tasques: la programació de la interfície d'usuari (la més extensa) i el disseny de les bases de dades. La interfície serà web i tindrà com a requeriments

⁷ <http://ask.libreoffice.org/en/question/18201/pdf-import-does-not-show-in-extension-manager/>

⁸ https://www.youtube.com/watch?v=MOa_qDY3q3I

⁹ <https://debianhelp.wordpress.com/2013/11/30/to-do-list-after-installing-linux-mint-17/>

l'equilibri entre la seva senzillesa d'ús i funcionalitat. El fet que sigui web, ha de possibilitar l'accés des de qualsevol terminal.

L'etapa d'implementació es centrarà en la instal·lació del programari en l'ordinador servidor (essencialment el SGBD i el servidor web) i la posada en marxa de la interfície d'usuari en el servidor web. Aquesta acció també requerirà formació als professionals i la generació de manuals d'ús.

Com s'ha indicat anteriorment, a partir d'unes primeres reunions amb els professionals del CEE i del propi anàlisi de viabilitat, s'han identificat quines són les necessitats específiques en quan a servei de gestió de la informació.

Com s'ha comentat, tota la informació es recull dins de fitxers de processadors de textos que cada usuari gestiona individualment. Això dificulta enormement l'accés i la cerca d'informació, així com compartirla entre els diversos professionals. Els diversos usuaris, segons el seu rol, caldria que tinguessin accés ràpid i senzill a les dades. No haurien de requerir demanar als companys que els hi subministressin la informació (que això és impossible si una persona ja no treballa en el centre). A través d'una interfície senzilla, els usuaris segons el seu rol haurien d'accedir immediatament a les dades. Això es pot assolir mitjançant un sistema de tractament modern de dades com és la utilització de BD i SGBD. La implementació d'aquest sistema, doncs, pot cobrir les necessitats actuals però alhora també n'ha de cobrir dues més de recurrents: complir amb el marc legal i minimitzar el cost. En aquest cas, el primer fa referència a la LOPD. En aquest aspecte i donat el tipus de dades que es gestionen, entre d'altres, caldrà declarar l'arxiu de dades o garantir la traçabilitat d'accés a les mateixes. Això es podrà assolir juntament amb l'assessorament de l'expert en aquesta matèria de què disposa el CEE i el tutor per part de la UOC, amb experiència professional en el tema. Per cobrir la necessitat del segon aspecte, l'ús de PL tant per l'SGDB com per l'ús de programari propi de gestió (que es desenvoluparà en el marc d'aquest projecte) realitzat també a partir d'eines lliures (en aquest cas, Netbeans [20] o Eclipse [21]) ho garanteixen.

Finalment, els dos factors que es constataven en l'anterior apartat (que els usuaris no són experts en informàtica i que alhora, no els cal un coneixement profund de la funcionalitat dels diversos programes) també apliquen en aquest.

A partir de l'anterior anàlisi de necessitats i el de viabilitat, es proposa específicament la implementació dels següents serveis:

- Gestió de la informació
 - Llistat d'alumnes amb les següents dades: nom i cognoms, data naixement, adreça, població, telèfons, noms pare i mare, correu electrònic, any entrada al centre, educació compartida o no, tutora.
 - Diagnòstics de les discapacitats dels alumnes: diagnòstic, professional o centre on s'ha fet i data.
 - Medicacions: medicació, qui l'ha receptat i qui fa el seguiment
 - Registre d'assistència.
 - Horaris de les aules i mestres.
 - *Informes*. Manca definir l'abast d'aquest punt.

- Registre d'incidències.
- Reunions: data, hora, assistents, lloc, temes tractats, acords, propera reunió.
- Normes de convivència i funcionament del Centre.
- *Programes individualitzats*. Manca definir l'abast d'aquest punt.
- **Bloc educatiu**. Es pretén crear un bloc educatiu però no amb una finalitat externa (pel públic extern) sinó com una eina de treball pels alumnes del centre. Es pretén que els alumnes puguin crear continguts i eventualment, mostrar-los als seus pares, amics, etc. Es tracta de crear un servei extremadament simple. Per aquest servei en concret, el qual es planifica com un contenidor de dades les quals no seran objectes de la Llei de Protecció de Dades i d'acord amb l'estudi de viabilitat, s'optarà per instal·lar-lo en l'ofert servei extern¹⁰ mencionat en el mateix estudi de viabilitat. El principal motiu és que cal que el servidor que l'allotgi ofereixi servei a Internet (i no és el cas del servidor intern d'AJEM) i en segon lloc, que el servei és gratuït. Per aquest servei de dades, s'usarà un servei de bloc adaptat que va programar l'autor d'aquest màster amb anterioritat.
- **Emmagatzematge de fotografies**. Per resoldre o mitigar l'inconvenient de l'emmagatzematge de fotografies comentat en la descripció del sistema actual, es proposen dues accions:
 - Desenvolupar una eina que automatitzi un procés de selecció i/o reducció de mida de fotografies.
 - Implementar un manual de bones pràctiques. Actualment, des de gerència es comenta el tema de viva-veu als professionals del CEE, però amb escàs èxit. La proposta és que els usuaris continuïn desant les fotografies en les seves màquines locals i l'eina anterior en desí còpies a la màquina servidor amb mida optimitzada. I les fotografies emmagatzemades en la màquina servidor estarien excloses del servei de còpia de seguretat (ja hi hauria un pseudo-còpia).

Adicionalment, AJEM planteja adquirir un nou disc dur per la màquina servidor que disposen. L'autor d'aquest projecte col·laborarà en aquesta acció la qual permetrà disposar de més espai per emmagatzemar fotografies.

En conseqüència, es programarà una interfície web que integrarà tota la gestió de la informació referents als punts anteriors. Aquesta interfície actuarà d'enllaç entre els professionals i les dades emmagatzemades al servidor. S'hi accedirà des de qualsevol navegador de qualsevol ordinador del centre (d'acord amb l'estudi de viabilitat serà un servei exclusiu d'intranet) i caldrà autenticar-se per accedir-hi. Es preveuen quatre rols d'usuari: usuari normal, responsable d'aula, director de centre i administrador.

La SGBD usada serà MySQL Community Edition [22]. Juntament amb PostgreSQL [23] són les dues solucions lliures més usades actualment. Les dues són perfectament viables per aquest projecte. Una de les diferències entre ambdós productes és que PostgreSQL està lleugerament més optimitzat per grans volums de dades, mentre que MySQL ho està per rapidesa en les consultes. I en aquest cas, una clara característica és el baix volum de dades a emmagatzemar.

Les dades de la base de dades s'integraran dins del sistema actual existent de còpies de seguretat.

¹⁰ necessito.cat

Com en l'apartat anterior, finalment, caldrà incloure els continguts d'aquest apartat en el llibre blanc de política d'IT del centre, en el manual i realitzar-ne formació bàsica als professionals del CEE.

2.1.3.- Resum dels requeriments

Lista de requeriments

- Política informàtica del CEE basada en Programari Lliure.
- Configuració dels ordinadors clients interns.
 - Sistema operatiu.
 - Paquet ofimàtic.
 - Editor d'imatges i editor de so.
 - Reproductor i còdecs multimèdia
- Creació de servei de gestió de la informació.
 - Gestió de la informació del CEE.
 - Bloc educatiu.
 - Gestió de fotografies.
- Manual d'ús.
- Formació als professionals del CEE.

Informes requerits i terminis

El projecte s'inicia el 26/02/15. A nivell organitzatiu, aquest inclou dos sub-projectes: les Pràctiques Externes (PrExt) i el Treball Final de Màster (TFM). Les PrExt finalitzen el 10/06/15 i el TFM ho fa el 17/06/15.

- Planificació del projecte (PrExt). Client: UOC i CEE. Termini: 11/03/15.
- Anàlisi de viabilitat i estudi del sistema. Client: UOC. Termini: 11/03/15.
- Disseny del sistema. Client: UOC. Termini: 15/04/15.
- Desenvolupament i implementació. Client: UOC. Termini: 27/05/15.
- Política d'IT del CEE. Client: CEE. Termini: 10/06/15.
- Manual bàsic d'ús dels recursos IT del CEE. Client: CEE. Termini: 10/06/15.
- Informe final del projecte (PrExt). Client: UOC i CEE. Termini: 10/06/15.
- Memòria TFM i defensa. Client: UOC. Termini: 17/06/15.

Recursos del projecte. Equip.

- Carles Torras Font, carlestorras@uoc.edu. Executor / alumne del màster.
- Anna Maria Valls Sastre, avalls26@xtec.cat. Tutora per part del CEE / Directora del CEE.
- Josep Maria Garcia, administracio@ajem.cat. Supervisor per part del CEE / Gerent d'AJEM.
- Víctor Robert Roca, vrobertr@uoc.edu. Tutor per part de la UOC.
- David García Solórzano, dgarciaso@uoc.edu. Professor responsable per part de la UOC / Responsable de PFM en l'àmbit de Direcció de Sistemes d'Informació.

Prioritats d'organització

- Respectar la Llei de Protecció de Dades.

- No comprometre els serveis actuals en funcionament.
- Aconseguir els objectius descrits.
- Apropar-se a cost zero del projecte.
- Causar el mínim trastorn possible als professionals del centre.

Pressupost

Es pretén executar el projecte sense recursos econòmics. Tanmateix, si cal realitzar alguna petita inversió d'equips, es podrà estudiar.

2.2.- Especificació del pla de proves

2.2.1.- A1: Programari de base

Per circumstàncies actuals, es preveu que una de les màquines que hi ha al CEE no s'usarà durant l'execució del projecte. Aquesta situació serà convenientment aprofitada i s'usarà aquesta màquina com a banc de proves. Serà la primera on s'implementarà el nou programari de base i amb la qual es realitzarà una primera etapa de formació (juntament amb alguna altre màquina externa).

A partir d'aquí i progressivament (donat que tota la implementació l'haurà de fer una persona compaginant-la amb altres tasques), s'anirà estenent la implementació a la resta de màquines. Com que els usuaris ja hauran rebut una primera formació, quan tinguin el nou programari ja començaran a saber com usar-lo. També haurien de disposar d'un primer manual bàsic.

Finalment, després de tota la implementació, es realitzarà una segona formació de consolidació i resolució de dubtes (ja que els usuaris ja hauran començat a usar el nou sistema).

2.2.2.- A2: Serveis de gestió de la informació

Un cop enllestida la tasca de desenvolupament principal dels serveis i la seva instal·lació en la màquina servidor d'AJEM (processos totalment transparents al client) es procedirà a realitzar-ne proves juntament amb els professionals del CEE per realitzar els ajustos finals. Com que el client accedirà als serveis via web, no caldrà cap plataforma especial i hi podran accedir tant des del sistema actual o del nou. Donat el temps disponible d'execució del projecte, els recursos i el volum de desenvolupament a realitzar, es preveu que amb 2 o 3 iteracions, el producte es doni per lliurat definitivament.

3.- Arquitectura

La definició de l'arquitectura del sistema és el primer requeriment per definir els components. L'objectiu és generar la documentació concreta que serveixi per definir específicament el projecte.

Primerament es realitza un resum de la situació actual (detallada en la planificació del projecte) i tasques establertes com a punt de partida. Això es relaciona amb com s'estableix el disseny del projecte. A continuació, es defineixen els nivells d'arquitectura pels serveis que es crearan en el projecte (Nou Programari de base, Bloc Educatiu i Sistema de Gestió de la Informació que inclou el gestor de fotografies), es defineixen les normes i estàndards de disseny i construcció, i finalment s'especifiquen els subsistemes que en aquest cas i especialment pel disseny del bloc educatiu, són escassos donada la poca complexitat dels mateixos.

3.1.- Estat actual

El CEE disposa de 8 ordinadors, impressores i antenes receptores de WIFI repartits en 6 aules. Cada màquina disposa d'un programari diferent, en funció, sobretot, del seu origen. En quan a serveis, els usuaris del CEE disposen del servei de carpeta en xarxa sobre el qual es fan còpies de seguretat. Tots els ordinadors disposen de connexió a Internet, irregular i deficient. A banda dels recursos informàtics propis del CEE, aquest també es beneficia de la infraestructura comuna d'AJEM. A banda de la connexió a Internet ja mencionada (i del corresponent encaminador propietat del proveïdor de serveis d'internet -ISP-), la resta d'infraestructura informàtica com mostra la figura 4 és una màquina servidor (que l'únic servei que dóna és el de còpies de seguretat de fitxers mitjançant el programa *Cobian Backup II* i discs durs USB externs), un sistema d'alimentació ininterrompuda (SAI) i un commutador.

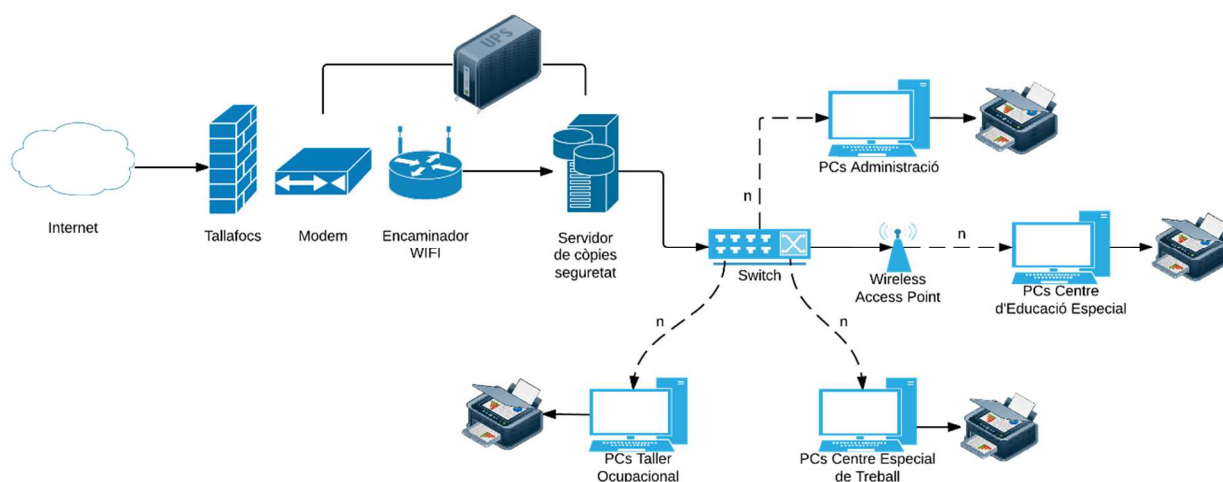


Figura 4. Infraestructura d'IT actual d'AJEM. Diagrama de components.

L'aparell de l'ISP realitza les funcions de tallafocs, mòdem i encaminador. La màquina servidor no està exposada a Internet i l'únic servei que dóna (el de còpies de seguretat) és intern.

3.1.1.- Especificacions actuals del maquinari d'ús comú d'AJEM

- Encaminador. Correspon a un model *Observe AW4062* de l'empresa Movistar. Realitza funcions de tallafocs, mòdem i encaminador¹¹.
- Màquina servidor. Correspon a un model *HP Proliant ML110 G7*¹² amb 2 Gb de memòria RAM i dos discs durs de 250 Gb. Disposa d'un sistema operatiu *Windows Server 2008 R2*. L'únic servei que proporciona és el de còpies de seguretat d'arxius, mencionat anteriorment. Per això, usa el programa *Cobian Backup II*¹³.
- Disc durs externs. Es disposa de 2 discs durs externs USB d'1 TB cadascun, els quals s'utilitzen per realitzar còpies de seguretat, tal i com s'ha descrit en l'anterior punt.
- Un sistema d'alimentació ininterrompuda (SAI). Correspon a un model de la marca *Netway Technology*.
- Commutador (switch). Es tracta d'un model *OvisLink EVO FSH24*¹⁴. Distribueix xarxa al Taller Ocupacional i Centre Especial de Treball per cable i a l'Escola (CEE) via una antena amplificadora (i els ordinadors del CEE disposen de dispositius receptors del senyal sense fils).

3.1.2.- Especificacions actuals de maquinari i programari del CEE

Tal i com mostra la figura 5, el CEE disposa de sis aules. Totes les aules disposen d'un ordinador menys dues que en disposen de dos. Així, hi ha un total de vuit ordinadors.

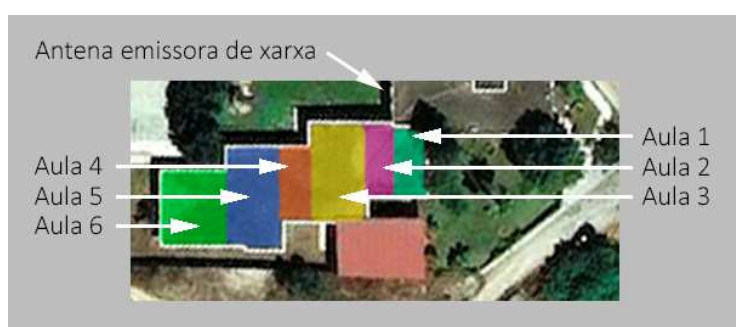


Figura 5. Plànol de situació de les aules del CEE

La següent taula mostra la relació d'aules i ordinadors, així com la seva identificació:

Nº PC	Nº Aula	Nom Aula	Nom màquina	Grup de treball
1	1	Lluna Nova B	BEEP1	AJEM
2	2	Lluna Nova A	--PC2	AJEM
3	3	Lluna Plena	--PC3	AJEM
4	4	Lluna Plena	--PC	WORKGROUP
5	5	Eclipsi	--PC5	AJEM
6	5	Quart Creixent	--PC	WORKGROUP
7	5	Quart Creixent	--PC6	AJEM
8	6	Logopèdia	ab3782aa21d84af	GRUPO_TRABAJO

¹¹ <http://www.movistar.es/particulares/atencion-cliente/internet/adsl/equipamiento-adsl/routers/observa-aw4062/>

¹² http://h20564.www2.hp.com/hpsc/doc/public/display?docId=emr_na-c02793565

¹³ <http://www.cobiansoft.com/cobianbackup.htm>

¹⁴ http://www.ovislink-espana.com/es/4_4int_1_18/Productos/Switches/Fast-ethernet/Evo-FSH24/

Per cadascuna de les màquines del CEE es va comprovar quin programari especial tenia instal·lat (a part dels comuns¹⁵ com són algun reproductor d'àudio o paquet ofimàtic), quins controladors usaven per gestionar el maquinari que cada PC disposava i quin volum de dades d'usuari contenia. A continuació es mostra la seva descripció.

Ordinador 1 (BEEP1) - LLUNA NOVA B, 1/1

Programari

- Windows Vista Home Basic
- Programes especials:
 - TOCATOCA -> Medi Natural -> Fruites i Verdures
 - Aplicacions PIE -> DNEE -> Fruites i Verdures

Maquinari

- Intel Pentium Dual CPU 1.60 GHz 32 bit, 3071 MB RAM
- Disc dur, C (465 Gb, 103 Gb carpeta usuari)
- NVIDIA GeForce 7050 / NVIDIA nForce 610i (NVIDIA, 31/01/2013, 9.18.13.783)
- NVIDIA nForce Networking Controller (NVIDIA, 10/9/2007, 67.6.2.0)
- Ralink Turbo Wireless LAN Card (Ralink Technology, 11/5/2007, 2.0.4.0)
- HP Deskjet F2100 (DOT4USB). Impressora + Escàner (HP, 16/2/2007, 1.0.0.0)
- HP Deskjet F2100 (HP, 27/11/2006, 8.0.0.0)
- VIA High Definition Audio (VIA Technologies, 16/10/2007, 6.0.1.1400)

Ordinador 2 (--PC2) - LLUNA NOVA A, 1/1

Programari

- Windows 7 Home Premium SP1
- Programes especials: Teletubbies, Aplicacions PIE -> Peix peixet + Una mosca, PIPO

Maquinari

- Intel Core2 Quad Q8200 @ 2.33 GHz 32 bit, 4.0 GB RAM
- Disc dur, C (465 Gb, 67,1 Gb carpeta usuari)
- ATI Radeon HD 4350 (ATI Technologies Inc., 23/09/2009, 8.661.0.0)
- Controladora Gigabit Ethernet Atheros AR8131 PCI-E NDIS 6.20 (Microsoft, 01/04/2009, 1.0.0.4)
- Ralink RT61 Turbo Wireless LAN Card (Ralink Technology, Inc., 11/12/2006, 2.0.3.0)
- ATI High Definition Audio Device (ATI Technologies, 18/09/2009, 5.0.7000.9)
- Realtek High Definition Audio (Realtek Semiconductors Corp., 22/9/2009, 6.0.1.5943)
- HP Deskjet 1510 series (REST) -> USB (Impressora + Escàner) (HP, 16/12/2012, 30.0.1044.40289)

Ordinador 3 (--PC3) - LLUNA PLENA, 1/2

Programari

- Windows 7 Home Premium SP1 32 bit
- Programes especials: AraSaac, AraSuite

Maquinari

- Intel Core2 Quad Q8200 @ 2.33 GHz 64 bit, 4Gb RAM
- 1 unitat de DVD
- Disc dur, C (465 Gb, 179 Gb carpeta usuari)
- Disc dur, X per compartir <\\192.168.1.2>

¹⁵ S'entén per programari comú aquell que ja està previst d'instal·lar en el nou sistema i que ha estat identificat en la fase de planificació del projecte.

- Disc dur, Y Carolina [\\192.168.1.2](#)
- ATI Radeon HD4350 (ATI Technologies, 23/09/2009, 8.661.0.0)
- Atheros AR8131 PCI-E Gigabit Ethernet Controller NDIS 6.20 (Atheros, 04/09/2009, 1.0.0.14)
- Netgear WG311v3 54Mbps Wireless PCI Adapter (Marvell, 16/10/2007, 1.0.0.53)
 - Model WN2000RPT v2
- HP Deskjet F2400 (DOT4USB). Impressora + Escàner (HP, 17/9/2008, 1.0.0.0)
- HP Deskjet F2400 (HP, 10/12/2008, 8.0.0.0)

Ordinador 4 (--PC) – LLUNA PLENA, 2/2

Programari

- Windows 7 Home Premium SP1 64 bit
- Programes especials: -

Maquinari

- Intel Core2 Quad Q8200 @ 2.33 GHz 64 bit, 4 Gb RAM
- 2 unitats de DVD
- Disc dur, C (465 Gb, 13 Gb carpeta usuari)
- ATI Radeon HD4350 (ATI Technologies, 17/08/2009, 8.632.1.2000)
- Atheros AR8131 PCI-E Gigabit Ethernet Controller NDIS 6.20 (Microsoft, 01/04/2009, 1.0.0.4)
- TP-Link 150Mbps Wireless Lite N Adapter (TP-Link, 04/01/2010, 2.0.0.32)
- Dispositiu de High Definition Audio (Microsoft, 13/7/2009, 6.1.7600.16385)

Ordinador 5 (--PC5) – ECLIPSI, 1/1

Programari

- Windows 7 Home Premium SP1 32 bit
- Programes especials: HOFMANN, AntiPorn (N/R)

Maquinari

- Intel Core2 Quad Q8200 @ 2.33 GHz 32 bit, 4 Gb RAM
- 2 unitat de DVD (una no funciona).
- Disc dur, C (465 Gb, 52 Gb carpeta usuari)
- Disc dur, Z Lina [\\192.168.1.2](#)
- ATI Radeon HD4350 (ATI Technologies, 23/09/2009, 8.661.0.0)
- Atheros AR8131 PCI-E Gigabit Ethernet Controller NDIS 6.20 (Atheros, 04/09/2009, 1.0.0.14)
- Dispositiu inalámbric 802.11b/g extensible Realtek 8185 (Microsoft, 21/06/2006, 6.1053.1002.2006)
- HP Deskjet F2400 (DOT4USB). Impressora + Escàner (HP, 17/9/2008, 1.0.0.0)
- ATI High Definition Audio Device (ATI Technologies, 18/09/2009, 5.0.7000.9)
- Realtek High Definition Audio (Realtek Semiconductors Corp., 22/9/2009, 6.0.1.5943)
- HP Deskjet F2400 (HP, 10/12/2008, 8.0.0.0)

Ordinador 6 (--PC) - QUART CREIXENT, 1/2

Programari

- Windows 7 Home Premium SP1 32 bit
- Programes especials:

Maquinari

- Intel Core2 Quad Q6600 @ 2.4 GHz 64 bit, 4 Gb RAM
- 2 unitats de DVD
- Disc dur, C (465 Gb, 0.7 Gb carpeta usuari)
- NVIDIA GeForce 8800GT (Microsoft Corporation – WDDM v1.1) (NVIDIA, 14/05/2009, 8.15.11.8593)

- NIC de Gigabit de Ethernet PCI-E Realtek RTL8168B/8111B (NDIS 6.20) (Microsoft, 26/2/2008, 7.2.1127.2008)
- TP-Link 150Mbps Wireless Lite N Adapter (TP-Link, 04/01/2010, 2.0.0.32)
 - Model TPLink TL-WN722N
- Dispositivo de High Definition Audio (Microsoft, 19/11/2010, 6.1.7601.17514)

Ordinador 7 (--PC6) – QUART CREIXENT, 2/2

Programari

- Windows 7 Home Premium SP1 32 bit
- Programes especials: PIPO Ampliar pantalla

Maquinari

- Intel Core2 Quad Q8200 @ 2.33 GHz 64 bit, 4 Gb RAM.
- 2 unitats DVD
- Disc dur, C (465 Gb, 39.7 Gb carpeta usuari)
- Disc dur, Z e [\\192.168.1.2](#)
- ATI Radeon HD4350 (ATI Technologies, 23/09/2009, 8.661.0.0)
- NIC de Gigabit de Ethernet PCI-E Realtek RTL8168C(P)/8111C(P) (NDIS 6.20) (Microsoft, 26/2/2008, 7.2.1127.2008)
- D-Link AirPlus G DWL-G510 Wireless PCI Adapter (rev. C) (D-Link, 11/12/2006, 2.0.3.0)
 - Model DLINK ANT24-0600 2.46GHz 6dBi
- ATI High Definition Audio Device (ATI Technologies, 18/09/2009, 5.0.7000.9)
- Realtek High Definition Audio (Realtek Semiconductors Corp., 22/9/2009, 6.0.1.5943)
- HP Deskjet 1050 J410 series (USB) (impressora + escàners) (HP, 05/06/2010, 9.0.0.0)

Ordinador 8 (ab3782aa21d84af) – LOGOPÈDIA, 1/1

Programari

- Windows XP SP2 32 bit
- Programes especials: -

Maquinari

- Intel Pentium D CPU 2.80 GHz 32 bit, 896 MB RAM
- 1 unitat de CD
- Disc dur, C (186 Gb, 52,6 Gb carpeta usuari)
- SIS Mirage Graphics (SIS, 26/10/2010, 6.14.10.3930)
- NIC Fast Ethernet PCI Familia RTL8139 de Realtek (Microsoft, 01/07/2001, 5.398.613.2003)
- Ralink Turbo Wireless LAN Card (Ralink Technology, 12/6/2009, 1.2.8.0)
- Realtek AC'97 Audio (Realtek Semiconductor Corp., 24/9/2008, 5.10.0.6300)
- HP Deskjet 1050 J410 series (impressora + escàners)

3.1.3.- Tasques del projecte i enfocament del disseny

Tal i com està explicat en la planificació del projecte, aquest TFM inclou dues accions:

- A1. Establiment del programari de base necessari. Elecció del mateix (basat en Programari Lliure tant com sigui possible) i implementació.
- A2. Establiment de serveis de gestió de la informació. Posada en marxa de la BD i SGBD. Programació de la interfície d'usuari adaptada a les necessitats del CEE i implementació.

De les dues accions i d'acord amb la planificació efectuada se'n deriven diverses tasques (veure cronograma). D'aquestes, se'n defineixen tres de principals que constitueixen l'abast del disseny del projecte:

- T2.1. El disseny, creació i implantació del Bloc Educatiu del CEE.
- T2.2. (i T2.3). El disseny, creació i implantació del Sistema de Gestió de la Informació que inclou un gestor fotogràfic (a efectes de disseny, les dues tasques es poden unificar).
- T3. La implantació del nou programari de base en els ordinadors del CEE

El disseny del projecte s'ha realitzat a nivell de cadascuna d'aquestes tres tasques. Les dues darreres corresponen a desenvolupament de programari, mentre que la primera correspon a la implementació de programari lliure existent. Així, la metodologia seguida pel disseny de les dues darreres tasques correspon a l'establerta en la guia del TFM [1], abordant cadascuna de les etapes indicades. Sobre la tasca restant també la segueix però només parcialment, degut a la seva pròpia tipologia que és més simple.

3.2.- Definició de nivells d'arquitectura

Com s'ha esmentat, es realitza la definició dels nivells d'arquitectura per les 3 tasques principals del projecte: T2.1, T2.2 (serveis) i T3 (nou programari de base).

3.2.1.- Programari de base (T3) i nova infraestructura

Es comença per la tasca T3 degut a què és la més simple (no inclou tots els passos de la metodologia citada) i perquè és la que ajuda a tenir una visió global de tot el projecte.

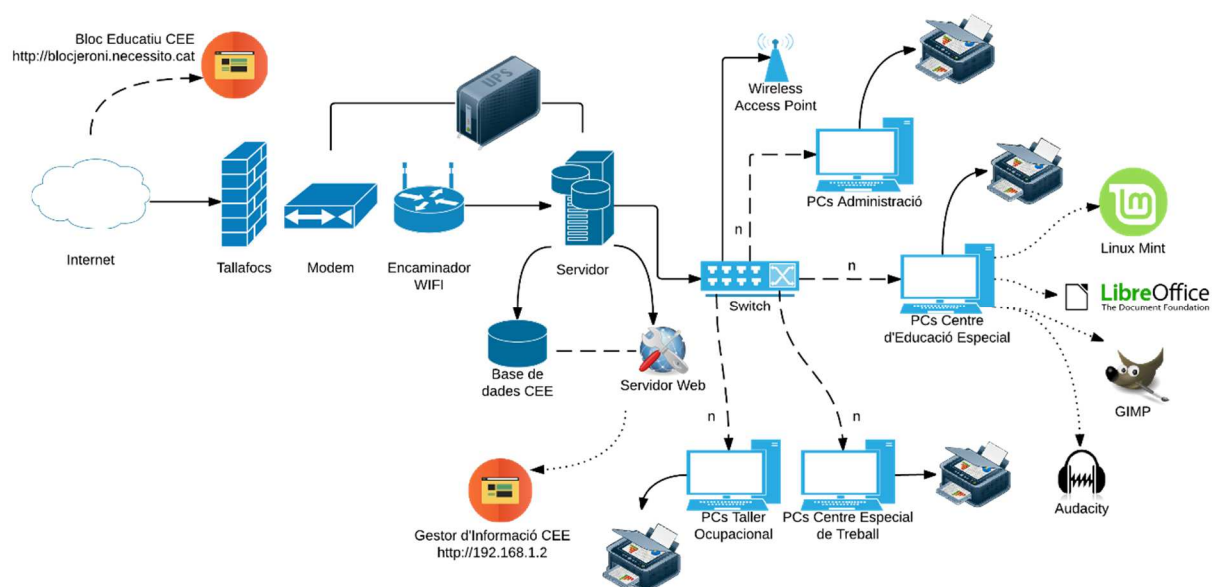


Figura 6. Infraestructura d'IT projectada d'AJEM. Diagrama de components.

La figura 6 mostra l'evolució projectada de la infraestructura d'IT en relació a la situació actual (figura 1). S'inclouen els dos nous serveis a generar (Bloc i Sistema de Gestió de la Informació) marcats com a serveis en taronja. El Gestor depèn de la màquina servidor d'AJEM a la qual se la proveirà amb un servidor web i una SGBD. El Bloc està situat en un servidor extern (el qual no s'indica). Pel què fa als ordinadors del CEE, s'indica que es projecta que tots treballin amb **Linux Mint** (SO per defecte), **LibreOffice** (paquet ofimàtic), **GIMP** (tractament d'imatges) i **Audacity** (tractament de so). Encara que

no està previst realitzar-ho en aquesta fase, sí que s'indica la prioritat que els ordinadors del CEE passin a estar connectats via cable i que el WIFI sigui un servei complementari (per mòbils, portàtils externs, etc.).

3.2.2.- Bloc Educatiu (T2.1)

El bloc educatiu és una eina especialment dissenyada como a recurs TIC per ser usat en les aules del CEE. És a dir, no té una finalitat orientada al públic extern, sinó a l'ús intern. Tanmateix, sí que es vol que les entrades efectuades puguin ser visualitzades pels alumnes amb la família, amics, etc., fora del centre.

Per tant, els requeriments principals de l'eina són els següents:

- Eina molt senzilla d'utilitzar que la puguin fer servir els alumnes i professionals del CEE.
- Que s'hi pugui connectar interna i externament. Al requerir-se servei extern, el servei no es pot instal·lar al servidor d'AJEM (només dona servei d'Intranet). Per tant, s'opta per usar uns serveis externs d'allotjament¹⁶. En aquest cas, aquesta solució també és viable donat que no s'introduiran continguts protegits per la LOPD.
- Ha de permetre els següents continguts: text i imatges.
- Que només els usuaris amb permisos (els professionals del CEE) puguin inserir entrades.
- Addicionalment, els professionals del centre no volen permetre comentaris externs.

Com que no es preveu que s'hi efectuïn un nombre molt elevat d'entrades, es proposa un sol bloc per tot el CEE i que les entrades es puguin filtrar per aula i data.

S'accedirà al bloc mitjançant la URL: *blocjeroni.necessito.cat*. Quan s'hi accedeixi, es visualitzarà una capçalera amb el títol i logotip del CEE, les darreres 10 entrades, un botó per inserir noves entrades, un botó per gestionar usuaris i un apartat per filtrar missatges per aula i data.

La pàgina per inserir noves entrades estarà formada per tres camps: el camp on s'introduirà el nom d'usuari/contrasenya que serà validat a l'enviar el nou comentari, el camp per introduir el text i el selector d'arxius per introduir la imatge (opcional).

La figura següent mostra el diagrama UML de components pel servei del Bloc Educatiu (T2.1) que cal crear en aquest projecte respecte l'acció A2.

¹⁶ *necessito.cat*

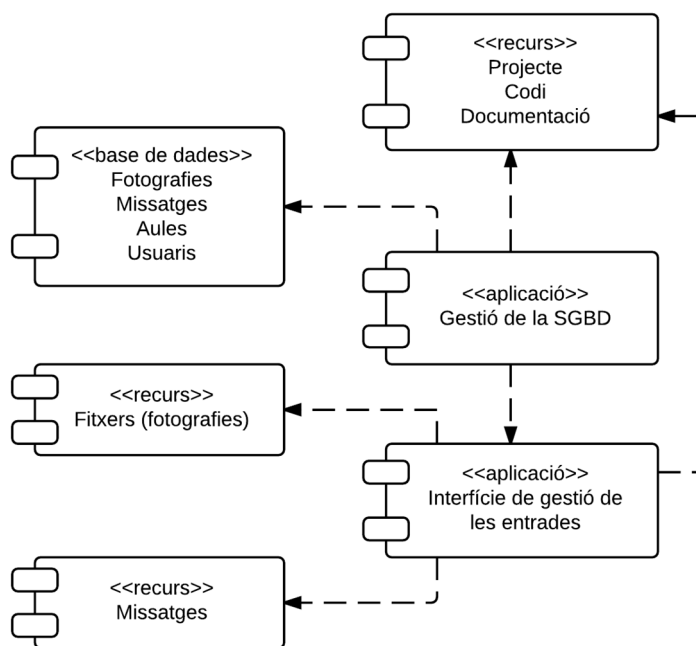


Figura 7. Diagrama UML de components del Bloc Educatiu.

S’identifiquen tres recursos. Els dos que constitueixen la font principal d’entrada d’informació: els fitxers de fotografies i els missatges i el relatiu a la sortida que correspon a la documentació del projecte: el codi i el manual d’ús i material de formació als professionals del CEE. Un dels aspectes a comentar d’aquest diagrama és la no presència ni d’un sistema de control de versions ni de còpies de seguretat. El primer no hi és perquè en aquest cas no cal (el servei extern ja el proveeix). El segon no hi és perquè tot i que cal, a l’instal·lar el servei en una infraestructura externa, aquesta ja té el seu sistema de còpies de seguretat.

També s’inclou la base de dades que conté la informació que cal que persisteixi i dues aplicacions: la que gestiona la base de dades i la que constitueix la interfície de l’usuari. Sobre la base de dades, en el diagrama s’han inclòs quatre conceptes: missatges i fotografies i aules i usuaris. Les aules i els usuaris no s’identifiquen com a recurs, ja que constitueixen la informació d’entrada rellevant del bloc.

Un cop definit el diagrama de components es defineixen les interfícies entre ells per tal d’obtenir l’arquitectura lògica del sistema. La següent figura mostra el diagrama anterior amb les interfícies detallant els processos de comunicació.

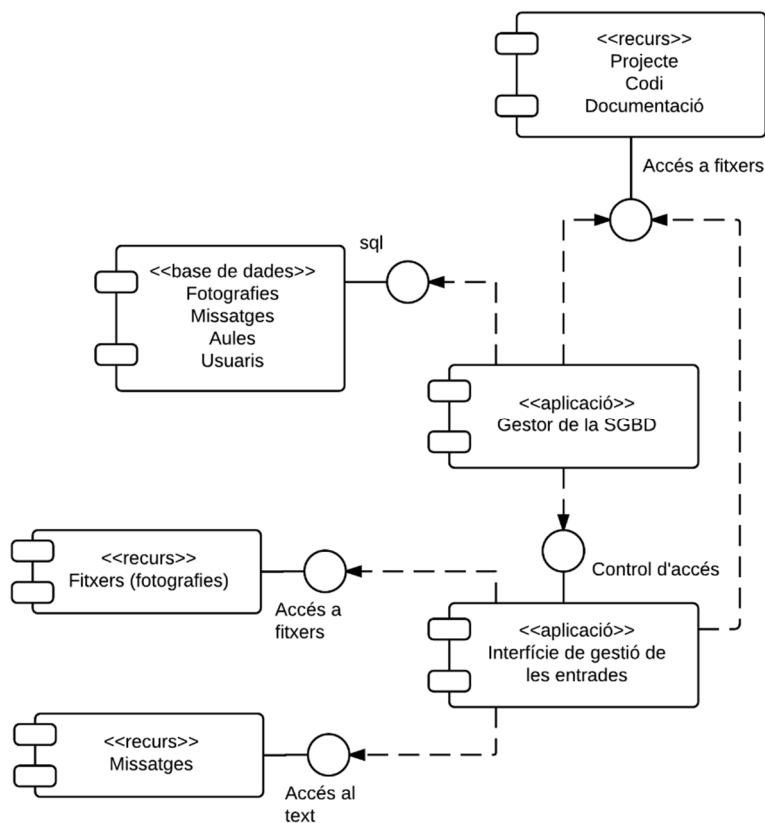


Figura 8. Diagrama UML de components amb interfícies del Bloc Educatiu.

Per cadascun dels components i per tal de detallar l’especificació de les interfícies identificades, es generen les següents targetes CRC:

Projecte, Codi, Documentació	
Establir tota la documentació del subprojecte.	-
Contenir tot el codi del projecte i les diverses versions que se n’han creat.	
Contenir el manual d’usuari del funcionament del bloc.	
Contenir les presentacions per tal de realitzar la formació sobre l’ús del bloc.	

Fotografies, missatges, aules, usuaris	
Contenir els enllaços de les imatges.	-
Contenir els missatges introduïts al bloc.	
Contenir els usuaris que poden introduir missatges i gestionar el bloc.	
Contenir el nom de les aules del CEE, que són les titulars de les entrades al bloc.	

Gestor de la SGBD	
Interacció amb la base de dades del bloc amb instruccions: CREATE, DELETE, INSERT, SELECT i UPDATE.	Accés als documents de subprojecte. Accés a la base de dades mitjançant instruccions SQL. Accés a la identitat de qui li demana realitzar operacions.

Interfície de gestió	
<p>Amb accessos validats com a usuaris:</p> <ul style="list-style-type: none"> • Introduir missatges. • Introduir una o més fotografies en cada missatge. • Possibilitat d'especificar títols a les fotografies • Eliminar els missatges creats. <p>Amb accessos validats com a administradors:</p> <ul style="list-style-type: none"> • Totes les operacions dels usuaris. • Crear, modificar o eliminar aules. • Crear, modificar o eliminar usuaris. • Eliminar missatges de qualsevol usuari. 	<p>Accés als documents del Projecte.</p> <p>Accés a les fotografies carregades pels usuaris.</p> <p>Accés als missatges introduïts pels usuaris.</p>

Fotografies	Missatges
Fotografies carregades pels usuaris -	Missatges carregats pels usuaris -

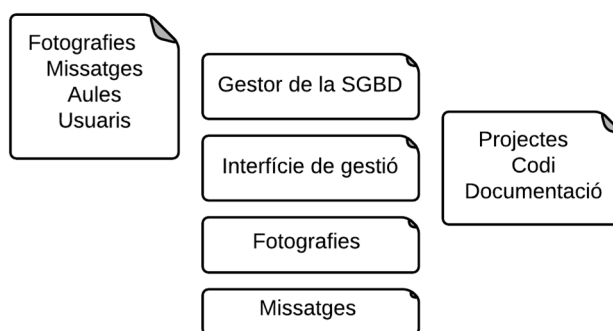


Figura 9. Diagrama de situació de targetes CRC pel Bloc Educatiu

3.2.3.- Sistema de Gestió de la Informació (T2.2)

El Sistema de Gestió de la Informació és una eina especialment dissenyada com a recurs TIC per ser usat com a administrador de les principals dades del CEE. És un servei intern que conté dades molt sensibles i objecte de la Llei de Protecció de Dades (LOPD).

Els requeriments principals de l'eina són els següents:

- Eina relativament senzilla d'utilitzar que la puguin fer servir els professionals del CEE per gestionar les dades del CEE.
- Per donar compliment a la LOPD i donades mancances actuals que no fan viable l'accés extern, es projecte que serà només d'ús intern. Tanmateix, cal dissenyar la plataforma perquè tècnicament sí que estigui preparat per l'ús extern. Per això, es desenvoluparà amb una interfície web.
- Ha de permetre els següents continguts: text i enllaços a arxius.
- Accés mitjançant autenticació i traçabilitat de les diverses accions efectuades, per quin usuari i quan. Donat el volum d'informació i tipus (baix) no es discriminarà per tipus d'informació

accedida (la informació es tractarà com un tot). En aquest cas, s'estant gestionant dades clíniques dels alumnes (diagnòstics, medicació establerta, etc.) les quals estant considerades de nivell alt per la LOPD i cal gestionar-les com a tal. Així, es traçaran tots els accessos a aquesta informació detalladament per si cal realitzar alguna comprovació al respecte.

- El Sistema de Gestió de la Informació ha de contenir dades sobre els professionals del CEE (identificació bàsica i categories), alumnes (identificació complerta, diagnòstic mèdic, medicacions, etc.), aules i definició de classes anuals, normes de centre, incidències, informes, reunions i assistència.
- El Gestor Fotogràfic ha de carregar fotografies, realitzar-ne un tractament bàsic: canvi de nom afegint el *timestamp* i reduint-li la mida a un valor òptim de qualitat i espai de disc, i desar-la sota etiquetes lliures, *timestamp* i aula.

S'accedirà al gestor mitjançant una IP interna. Quan s'hi accedeixi, caldrà que l'usuari completi la seva autenticació i un cop validada, en funció del rol que tingui, accedirà a un menú principal amb les opcions que li correspongui. El programa anirà desant les diverses accions que realitzi.

La figura següent mostra el diagrama UML de components pel servei del Sistema de Gestió de la Informació (T2.2) que cal crear en aquest projecte respecte l'acció A2.

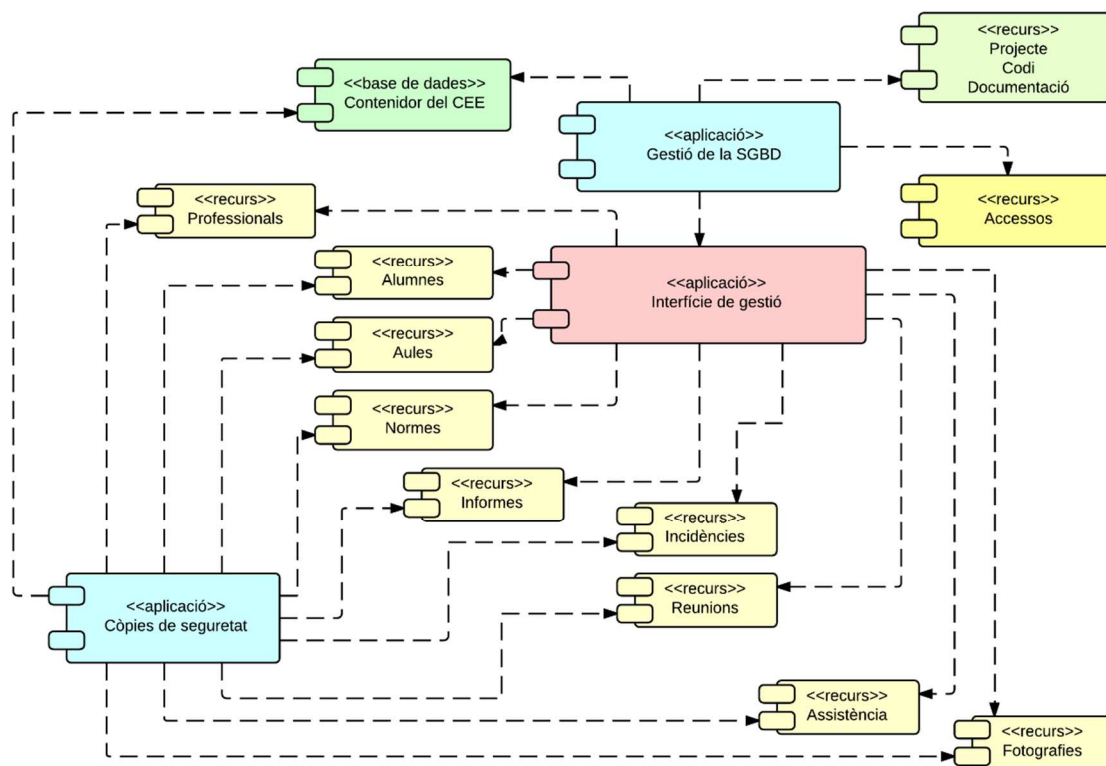


Figura 10. Diagrama UML de components del Sistema de Gestió de la Informació.

La complexitat del diagrama és major que en el Bloc Educatiu, malgrat que l'estructura és similar. El que fa augmentar la complexitat és la quantitat de recursos que cal gestionar. D'altra banda, la

informació que es maneja en aquest servei és més sensible i requereix més garanties. I en aquest cas, sí que es té en compte específicament un sistema de còpies de seguretat.

Un cop definit el diagrama de components es defineixen les interfícies entre ells per tal d’obtenir l’arquitectura lògica del sistema. La següent figura mostra el diagrama anterior amb les interfícies detallant els processos de comunicació.

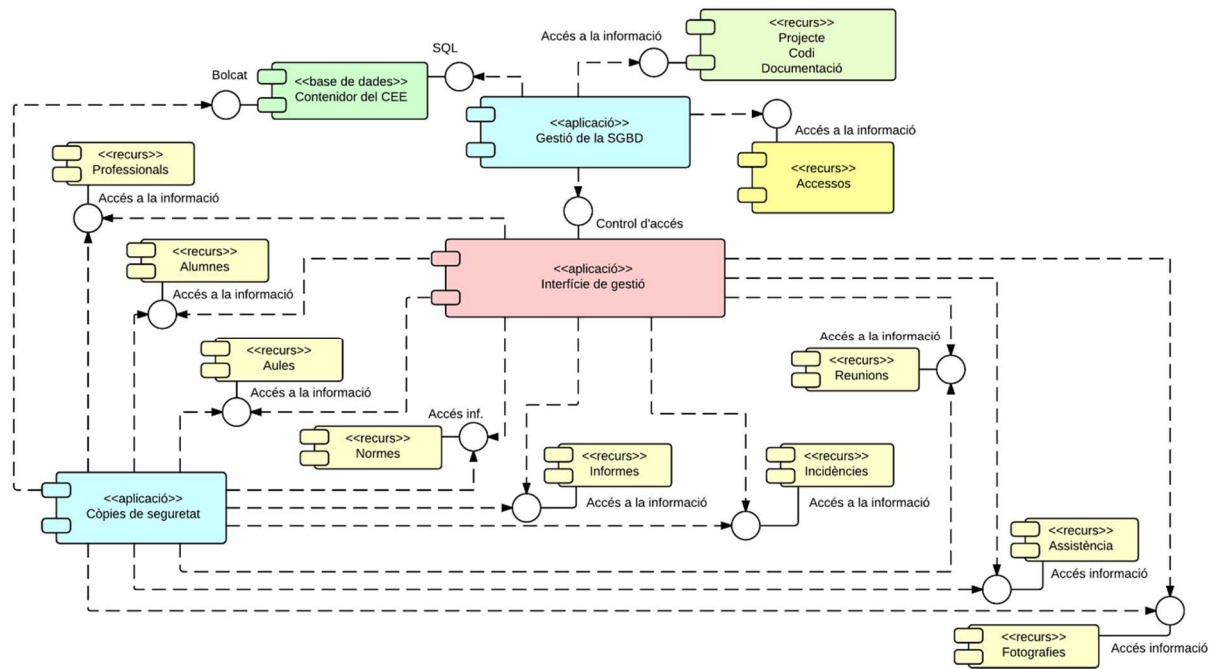


Figura 11. Diagrama UML de components amb interfícies del Sistema de Gestió de la Informació.

Per cadascun dels components i per tal de detallar l’especificació de les interfícies identificades, es generen les següents targetes CRC:

Projecte, Codi, Documentació	
Establir tota la documentació del subprojecte.	-
Contenir tot el codi del projecte i les diverses versions que se n’han creat.	
Contenir el manual d’usuari del funcionament del bloc.	
Contenir les presentacions per tal de realitzar la formació sobre l’ús del bloc.	

Contenedor del CEE	
Contenir tota la informació del CEE definida	-

Gestor de la SGBD	
Interacció amb la base de dades del bloc amb instruccions: CREATE, DELETE, INSERT, SELECT i UPDATE.	Accés als documents de subprojecte. Accés a la base de dades mitjançant instruccions SQL. Accés a la identitat de qui li demana realitzar operacions.

Interfície de gestió	
<p>Amb accessos validats com a tutor:</p> <ul style="list-style-type: none"> • Carregar fotografies i eliminar les pròpies. • Crear, modificar i eliminar reunions pròpies. • Crear, modificar i eliminar incidències dels alumnes de les seves classes. • Crear i eliminar faltes d'assistència dels alumnes de les seves classes. • Crear, modificar i eliminar informes dels alumnes de les seves classes. <p>Amb accessos validats com a director:</p> <ul style="list-style-type: none"> • Totes les operacions dels usuaris i de d'objectes propis o aliens. • Crear, modificar i eliminar professionals. • Crear, modificar i eliminar alumnes. • Crear, modificar i eliminar aules. • Crear, modificar i eliminar classes a partir dels alumnes, aules i professionals. • Crear, modificar i eliminar normes. 	<p>Accés als diversos recursos:</p> <ul style="list-style-type: none"> • Professionals • Alumnes • Aules • Normes • Incidències • Assistències • Informes • Reunions • Fotografies

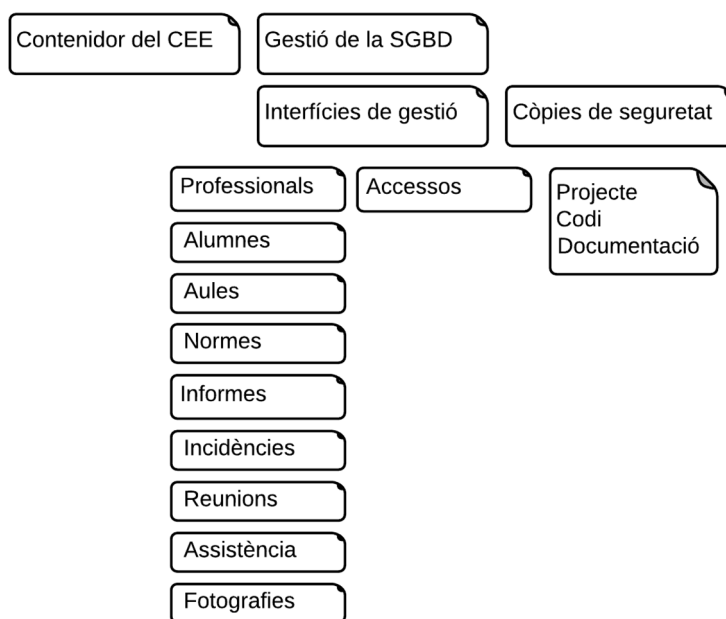
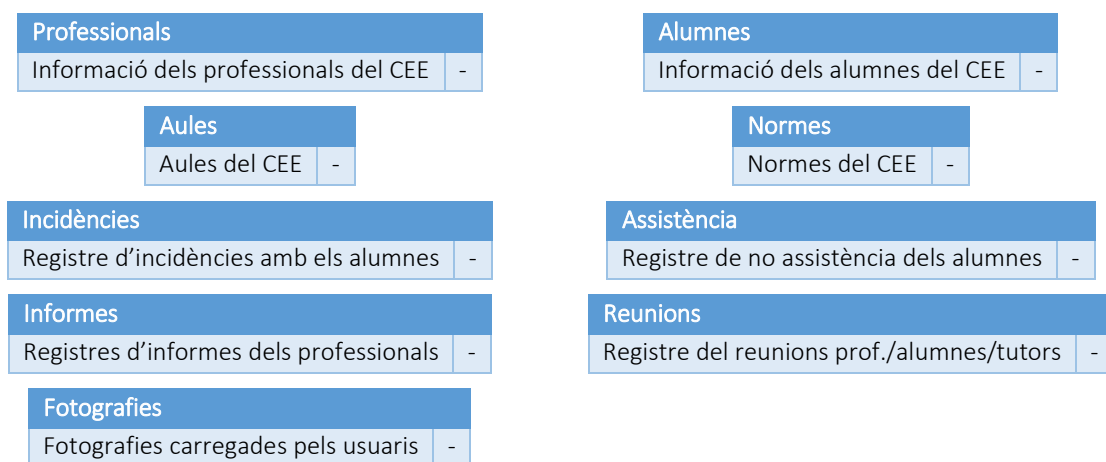


Figura 12. Diagrama de situació de targetes CRC pel Sistema de Gestió de la Informació

3.3.- Especificació d'estàndards, normes de disseny i construcció

A continuació s'estableix l'especificació d'estàndards, normes de disseny i construcció per tots els materials generats en aquest projecte. Per aquest apartat i llevat que no s'indiqui el contrari el l'apartat corresponents, cadascuna de les especificacions són a utilitzar per tots els sub-projectes.

- Documentació. Tots els documents de text seran presentat en format PDF, tant els documents de treball, esborranys com lliuraments finals. Els comentaris o correccions que s'hagin de realitzar en els documents es realitzarà mitjançant les eines de revisió i comentaris de PDF que per exemple estant disponibles en l'Adobe Reader (programa gratuït que es pot descarregar d'Internet). Això simplifica l'ús de formats i la justificació és que l'autor del projecte és únic i no es preveu treball col·laboratiu. La única excepció són els documents identificats com a lliurables al CEE (per exemple la política d'IT del CEE), que donat que podran ser modificats posteriorment pel CEE per fer-ne noves versions, es lliuraran també en el format lliure .odt.
- Presentacions. Els documents de presentació seran generats per poder realitzar la formació als professionals del CEE. Aquests es lliuraran en format PDF.
- Llibre d'estil. Els documents generats hauran de complir la normativa d'estil dels TFM de la UOC [24].
- Idioma. L'idioma oficial per tot el projecte és el català.
- Llenguatge de programació. S'estableixen dos llenguatges de programació principal per aquest projecte: HTML i Java. El primer és l'estàndard a usar per generar pàgines web que calen per les diverses interfícies. El segon és un dels llenguatges més usats actualment que té una gran potència i funcionalitat per tota la programació en general. Així, Java serà usat per programar *servlets*, subrutines, *javascrpts*, etc., i també serà incorporat en les pròpies pàgines HTML utilitzant etiquetes *jstl* i creant pàgines *.jsp*. Més endavant, es detalla més aquesta part.
 - Format de dades intercanviades. En la majoria dels casos, la informació intercanviada entre pàgines web i *servlets* es realitzarà a través de variables (formularis web, tractament *request/response* i usabilitat a través de *jstl*). Tanmateix, també es tindrà en compta el format *JSON* per intercanvi de cadenes d'informació.
 - Estils. En les pàgines web, la definició dels elements gràfics i estils s'especificarà, majoritàriament, mitjançant *CSS*.

Les especificacions de programari tècnic a usar i normes de disseny en quan al desenvolupament es detallen en la secció 2.2.

3.4.- Identificació dels subsistemes

La complexitat en els subprojectes del Bloc Educatiu i del Sistema de Gestió de la Informació és de nivells diferents. Mentre que en el primer cas la complexitat és baixa, en el segon és pot classificar de moderada.

Pel cas del Bloc Educatiu, es poden identificar tres subsistemes (gestió de la informació, gestor del bloc i gestor de la documentació) atenent a una gestió de la funcionalitat tal i com mostra la figura 13.

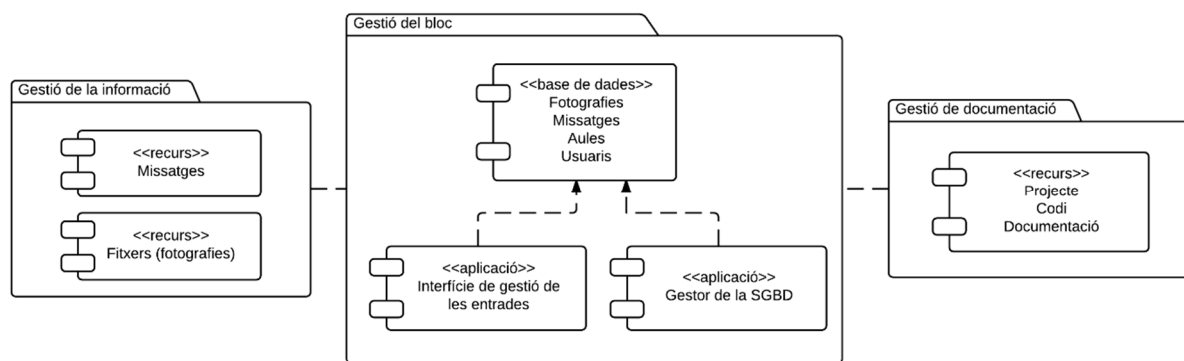


Figura 13. Diagrama de situació pel Bloc Educatiu

Pel cas del Sistema de Gestió de la Informació l'estructura és similar:

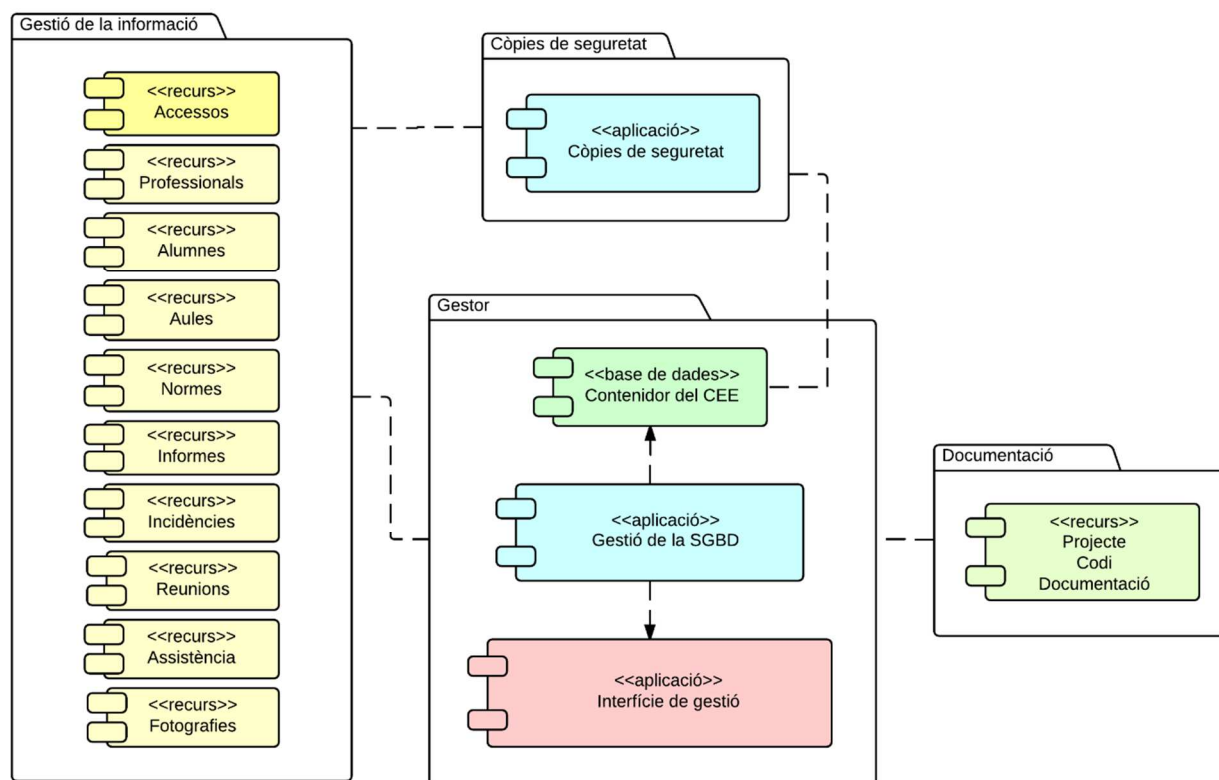


Figura 14. Diagrama de situació pel Sistema de Gestió de la Informació

4.- Casos d'ús

Una vegada identificats els subsistemes, es revisen els casos d'ús establerts en la fase d'anàlisi i que determinen les operacions que hauran d'implementar les interfícies de cada un d'ells.

4.1.- Revisió de casos d'ús per subsistema

Per a cadascun dels sub-projectes, es defineixen a continuació els subsistemes i actors que hi intervenen, així com els missatges que intercanvien els objectes.

4.1.1.- Bloc educatiu (T2.1)

El bloc ha de permetre, essencialment, introduir missatges a usuaris autoritzats, els quals podran incorporar una o més fotografies opcionalment. Aquestes, caldrà que siguin tractades automàticament per mostrar-les en un format adequat al Bloc (cal preveure que l'usuari les usi sense previ tractament). Els missatges podran ser eliminats per l'autor o per l'usuari, però les fotografies es mantindran desades (com a còpia de seguretat).

Així, es mostraran les entrades amb la següent informació: missatge, fotografia/es (opcional), peu de fotografia (opcional), data i hora de l'entrada, identificador de la mateixa i aula a la que correspon. És a dir, cada entrada estarà associada a una aula i no a l'usuari. Això és així ja que es vol que el protagonisme el tinguin l'entrada i l'aula. A més, es preveu que una aula pugui tenir involucrat més d'un professional (usuari del bloc) i que aquests puguin canviar. Així, un usuari pot ser eliminat (per exemple, al causar baixa al CEE), però el missatge corresponent a una aula persisteix (el missatge mai podrà quedar orfe d'aula).

Un o més usuaris podran ser administradors, els quals tindran possibilitat de gestionar les aules, els usuaris i els missatges de tots. Els usuaris no administradors, només podran crear entrades i eliminar les que hagin creat.

Les entrades s'han de poder filtrar per aula i data.

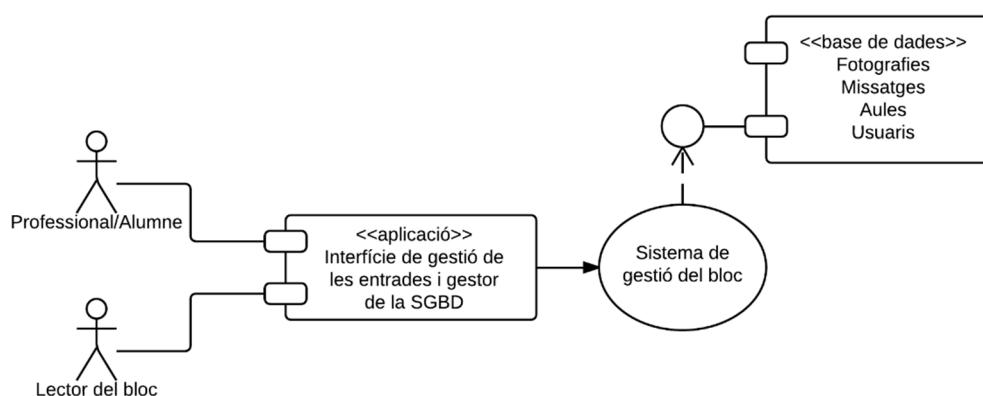


Figura 15. Diagrama UML de revisió de casos d'ús amb components implicats pel Bloc Educatiu

La figura 15 mostra el diagrama UML de revisió de cas d'ús amb els components implicats. En aquest cas, es defineixen dos actors principals que són qui realitza les entrades al bloc (professionals del CEE)

i els lectors del mateix. El sistema es dissenya a fi i efecte que no es requereixi que hi hagi un administrador del sistema, malgrat que algun professional del CEE, en algun moment puntual, hagi de realitzar alguna tasca. El fet que el bloc s'allotgi en un servidor extern proveït de còpies de seguretat, facilita aquest aspecte.

A continuació, s'identifiquen els missatges que intervenen en el bolcat de la base de dades al fitxer i els sistemes o objectes amb què s'intercanvien aquests missatges. Això condueix al diagrama de classes del subsistema, on es representen els objectes que hi intervenen i els seus atributs i mètodes.

Es requereix disposar d'una llista d'usuaris registrats que seran els que podran efectuar entrades al Bloc. Per simplificar el procés, no s'opta per la solució habitual de generar noms d'usuari i contrasenya, sinó només una clau que farà la funció dels dos. S'usarà encriptadament (protocol SHA-256). Aquesta paraula estarà lligada a un nom d'aula, que serà la que apareixerà com a autor del comentari. Les entrades al bloc requeriran el codi d'usuari, el text i opcionalment una fotografia. Quan s'enviïn les entrades, el sistema captarà automàticament la data i hora.

Per tant, el diagrama entitat/relació de la base de dades és el següent:

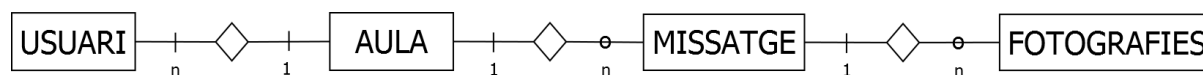


Figura 16. Diagrama entitat/relació del Bloc Educatiu

El model relacions de l'anterior diagrama ER, en aquest cas, és el següent:

```

AULA(nomaula)
USUARI(clau, nomaula, admin, descrip)
  on {nomaula} referencia AULA
MISSATGE(id, eltext, quan, nomaula)
  on {nomaula} referencia AULA
FOTOGRAFIA(idfoto, fotografia, idmissatge, titol)
  on {idmissatge} referencia MISSATGE
  
```

En relació a l'usuari, la clau i l'aula ja s'han definit anteriorment, i "admin" fa referència a si l'usuari és administrador o no. Si ho és, podrà gestionar altres usuaris i si no ho és, no. Les claus que identifiquen els usuaris seran encriptades segons el protocol SHA-256, per aquest motiu, se'ls dona una mida suficient.

A partir d'aquestes definicions, es pot generar l'esquemal SGBD. En primer lloc, es crea l'esquema que s'anomena *blocjeroni*. En segon lloc, es crea un nom d'usuari (*jeroni*) que s'associa amb l'esquema. En tercer lloc, s'afegeix la contrasenya a l'usuari (*jeroni*) i s'actualitza el sistema. I finalment, es generen les taules mostrades en l'annex 3. Aquestes taules es poden representar en la següent figura, corresponent al diagrama UML de classes (model relacional):

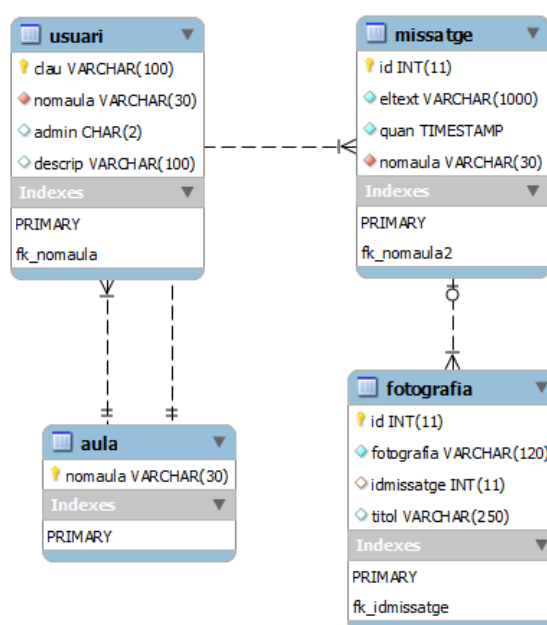


Figura 17. Diagrama UML de classes (model relacional) del Bloc Educatiu

4.1.2.- Sistema de Gestió de la Informació (T2.2)

El Sistema de Gestió de la Informació ha de permetre administrar les dades més rellevants del CEE, així com aprofitar la seva infraestructura per arxivar fotografies d'una manera eficient i que puguin ser també amb considerades pel servei de còpies de seguretat.

Es tracta d'un servei intern del CEE al qual s'hi accedirà a través d'una IP local (la del servidor), i mitjançant una primera autenticació mitjançant nom d'usuari i contrasenya. Els diversos professionals del CEE hi estaran donats d'alta amb rols diferents: administrador, director, tutor, usuari. Segons el grup a què pertanyin, podran realitzar determinades accions.

El gestor tindrà tres apartats principals: administració del CEE, administració de l'aula i gestor fotogràfic. En l'apartat d'administració de CEE (rol de director) es podran gestionar professionals, alumnes, normes i aules. En l'apartat d'administració d'aules (rol de tutor) es podran gestionar reunions, informes, incidències i faltes d'assistència.

El rol d'usuari permetrà gestionar fotografies en l'apartat del gestor fotogràfic. El rol de tutor permetrà gestionar reunions, informes, incidències i faltes d'assistència en l'administrador d'aules i totes les accions del rol d'usuari. El rol de director permetrà gestionar professionals, alumnes, normes i aules i totes les accions dels rols de tutor i usuari. El rol d'administrador tindrà les mateixes funcions que el de director però des d'una perspectiva tècnica que serà monitoritzada. Gestionar significa veure, introduir, modificar i eliminar entrades.

Per donar compliment a la LOPD, cada accés serà desat a la base de dades i la informació que contindrà serà nom d'usuari, data i hora i tipus d'acció/ns realitzada/es.

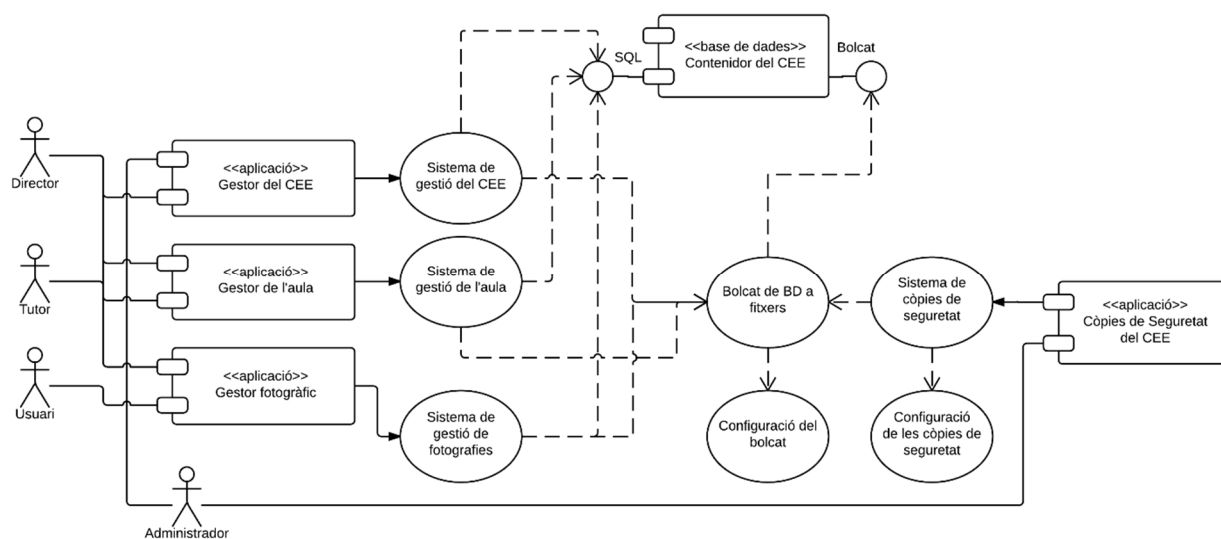


Figura 18. Diagrama UML de revisió de casos d'ús amb components implicats pel Sistema de Gestió de la Informació

La figura 18 mostra el diagrama UML de revisió de cas d'ús amb els components implicats. En aquest cas, es defineixen els quatre actors principals mencionats anteriorment (administrador, director, tutor i usuari). En aquets cas, sí que es consideren còpies de seguretat de les dades i fitxers. La gestió del mateix s'integra en l'actual sistema de còpies de seguretat (*Cobian Backup II*). L'administrador també actua sobre aquest aspecte però de la mateixa manera que actualment ja actua sobre el sistema actual (per tant, no s'afegeixen recursos al sistema).

A continuació, s'identifiquen els missatges que intervenen en el bolcat de la base de dades al fitxer i els sistemes o objectes amb què s'intercanvien aquests missatges. Això condueix al diagrama de classes del subsistema, on es representen els objectes que hi intervenen i els seus atributs i mètodes.

El disseny de la base de dades és un dels aspectes claus d'aquest gestor, ja que l'objectiu principal és, justament, gestionar les dades del CEE.

La definició de les dades que calia contenir en la SGBD i les seves relacions s'ha efectuat conjuntament entre els professionals del CEE i l'autor. La figura 16 mostra el diagrama entitat/relació del Sistema de Gestió de la Informació. Es fa palès que les dades giren entorn de dos grans entitats que són l'"alumne" i el "professional". Ambdós són una especialització de l'entitat "persona". A partir d'ells es genera un segon grup d'entitats que són les "classes", "incidència", "assistència", "informe" i "reunió". Llavors, existeix un tercer grup d'entitats que donen informació complementària que són la "població", "cursos", "aula", "norma", "fotografia" i les "categoria". Finalment, existeixen dues entitats més per controlar l'accés a la informació (qui i què): "accés" i "acció". Definides les entitats, es defineixen les relacions entre elles, especificant el caràcter optatiu o obligatori i si són relacions de 1:1, 1:n o n:m.

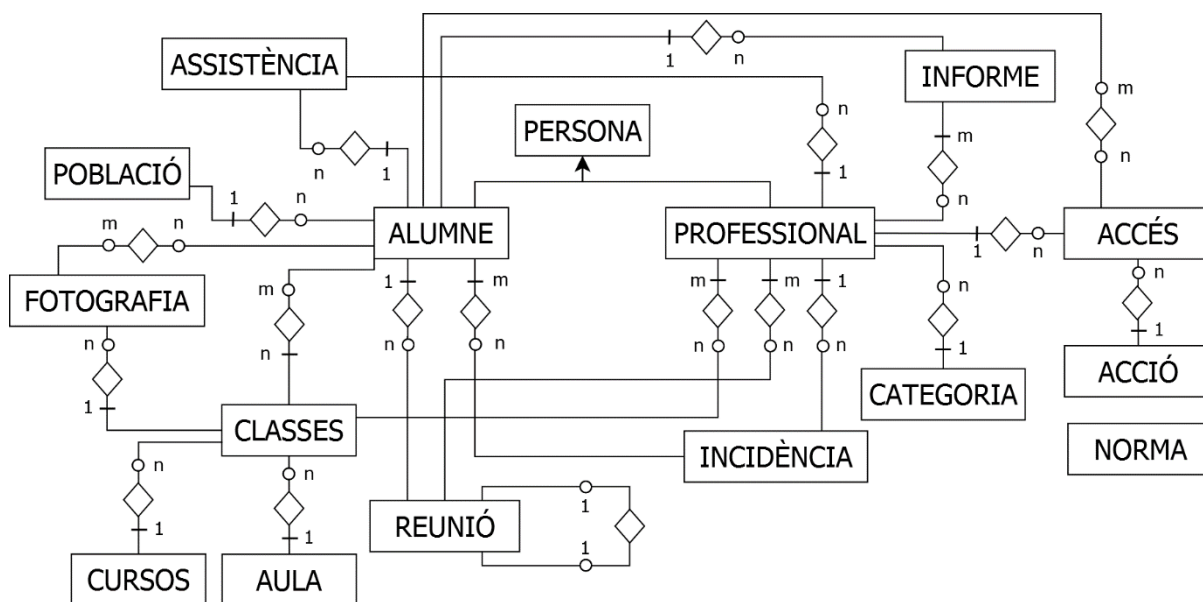


Figura 19. Diagrama entitat/relació del Sistema de Gestió de la Informació

A continuació, es mostra el model de relacions de l'anterior diagrama ER en el qual s'hi afegeixen totes les tuples de cada entitat (existeixen algunes modificacions respecte l'anterior diagrama que es comenten a continuació):

```

CATEGORIA(nomcategoria)
PROFESSIONAL(nif, nom, cognom1, cognom2, tutor, lkhorari, password, nomcategoria)
    on {nomcategoria} referencia CATEGORIA
MUNICIPI(id, nommunicipi, nomprovincia)
ALUMNE(nif, nom, cognom1, cognom2, nompare, nommare, nomparellapare, nomparellamare,
nomtutor, numtargetasanitaria, compartit, anyentrada, naixement, adreça, nommunicipi,
email, telefon1, telefon2, telefon3, telefon4, diagnostic1, diagnostic2, diagnostic3,
diagnostic4, diagnostic5, medicació1, medicació2, medicació3, medicació4, medicació5,
lkprograma)
    on {nommunicipi} referencia MUNICIPI
ASSISTENCIA(idassistencia, data, motiu, alumne, professional)
    on {alumne} referencia ALUMNE
    on {professional} referencia PROFESSIONAL
REUNIO(idreunio, datahora, altresassistents, lkcontingut, propera, nif)
    on {propera} referencia REUNIO
    on {nif} referencia ALUMNE
PROFREUNIONS(idreunio, nif)
    on {idreunio} referencia REUNIO
    on {nif} referencia PROFESSIONAL
AULA(nomaula)
CURSOS(curs)
CLASSES(nomaula, curs, lkhorari)
    on {nomaula} referencia AULES
    on {curs} referencia CURSOS
PROFCLASSE(nifprof, elnomaula, elcursaula)
    on {nifprof} referencia PROFESSIONAL
    on {elnomaula} referencia CLASSES
    on {elcursaula} referencia CLASSES
    
```

```
ALUMCLASSE(nomaula, nif, curs)
    on {nomaula} referencia CLASSES
    on {nif} referencia ALUMNE
    on {curs} referencia CLASSES
FOTOGRAFIA(id, enllaç, aula, etiqueta1, etiqueta2, etiqueta3, etiqueta4, data)
    on {nomaula} referencia AULA
FOTOALUMNE(id, nif)
    on [25] referencia FOTOGRAFIA
    on {nif} referencia ALUMNE
INFORME(idinforme, data, titol, lkinforme, alumne)
    on {alumne} referencia ALUMNE
PROFINFORMES(idinforme, nif)
    on {nif} referencia PROFESSIONAL
    on {idinforme} referencia INFORME
INCIDENCIA(idincidencia, nomincidencia, lkinforme, datahora, nif)
    on {nif} referencia PROFESSIONAL
INCIALUMNE(idincidencia, nif)
    on {nif} referencia ALUMNE
    on {idincidencia} referencia INCIDENCIA
NORMES(id, titol, lknorma)
ACCIO(nomaccio)
ACCES(id, timestamp, nif, nomaccio)
    on {nif} referencia PROFESSIONAL
    on {nomaccio} referencia ACCIO
ACCESALS(idaccesals, nifalumne)
    on {nifalumne} referencia ALUMNE
    on {idaccesals} referencia ACCES
```

En primer lloc, es constata que degut a relacions tipus n:m, s'han generat tres entitats més: "accesals", "alumclasse", "fotoalumne", "incialumne", "profinformes" i "profreunions". En segon lloc, per simplificar el disseny del sistema, s'ha eliminat la relació d'especialització alumne/professional. Això simplifica l'estructura i malgrat es disminueix lleugerament la robustesa del sistema, no la compromet en absolut.

Finalment, cal comentar l'entitat *alumne*. Aquesta entitat conté moltes tuples i algunes serien susceptibles de generar noves entitats i relacions. Tanmateix, la discussió amb els professionals del centre va aconsellar realitzar el disseny de la manera mostrada. En primer lloc, existeixen diversos camps implícits i explícits que referencien a persones (pare, mare, tutor, parelles, metges que realitzen diagnòstics, etc.). Aquests podrien ser una especialització de l'entitat *persona*. Tanmateix, es va constatar que en molts casos no es disposaria de la informació mínima que defineix *persona* (com el NIF) ni seria viable aconseguir-la. En quan al referent a diagnòstics i medicacions, també es va constatar que era molt difícil aconseguir tenir la informació en patrons. És a dir, que per cada alumne, el diagnòstic, que està relacionat amb la medicació i alhora a un metge i a un centre mèdic, implica una informació singular. A més a més, al no disposar d'informació sobre el metge i el centre mèdic, no feia aconsellable generar un seguit de noves entitats per relacionar-les convenientment. Alhora, la complexitat del sistema augmentaria. Per tot això i donats els recursos disponibles en aquest fase, es va decidir deixar-ho com a camp de text lliure. En següents versions i si la disponibilitat d'informació canvia, es milloraran aquests aspectes.

Cal també esmentar que de totes les entitats, n'hi una que conté dades pre-establertes i que s'han obtingut del web MuniCat [26]: l'entitat *municipi* que inclou els 947 municipis de Catalunya associats a la seva província.

Tenint en compte tot l'anterior, es generen les taules finals de la base de dades mostrades en l'annex 3. Aquestes taules es poden representar en la següent figura, corresponent al diagrama UML de classes (model relacional):

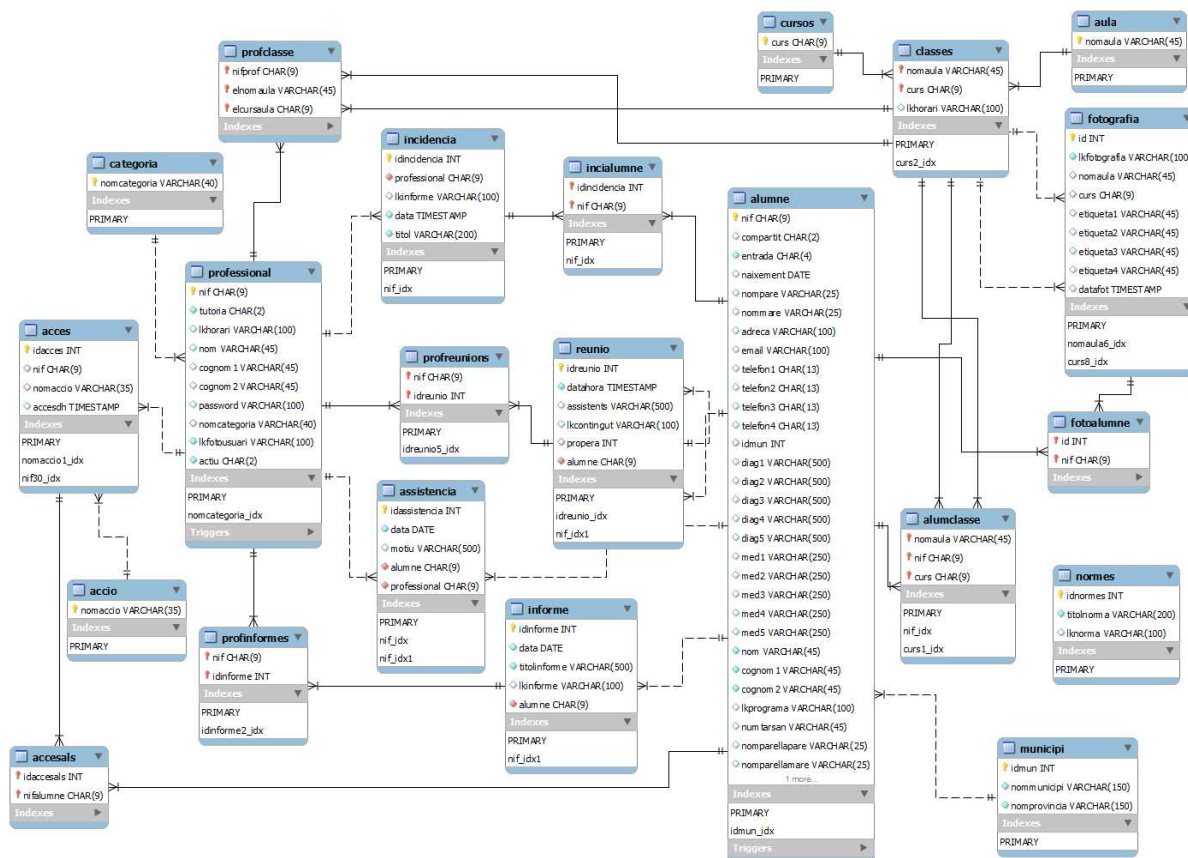


Figura 20. Diagrama UML de classes (model relacional) del Sistema de Gestió de la Informació.

4.1.3.- Programari de base (T3)

En seccions anteriors s'indicava el detall de l'inventari realitzat en relació al programari i maquinari existent actualment al CEE, un cop finalitzat. Dels resultats obtinguts, se'n desprenen els següents aspectes rellevants:

- Que no existeix cap maquinari o programari que d'una manera evident impedeixin la instal·lació d'un SO GNU/Linux i la resta de programari planificat.
 - S'ha comprovat que en algunes màquines hi ha algun programari instal·lat no identificat en la fase de planificació, susceptible de no poder ser instal·lat directament sobre un SO GNU/Linux. Tanmateix, després d'analitzar-ho amb els professionals del CEE, s'ha conclòs o que és prescindible o que es podrà instal·lar sota WINE.

- Que en la majoria d'ordinadors existeix un volum de dades d'usuari important (> 50 Gb). Després d'analitzar-ho amb els professionals del centre:
 - S'ha fet palès que es desconeix, en bona mesura, la informació emmagatzemada. És a dir, són contenidors d'arxius històrics que no estant inventariats, indexats, etc.
- Que en cada classe hi ha un ordinador principal usat pel tutor (el més crític) i a part, en dues classes, hi ha en cadascuna un altre ordinador el qual és usat pels alumnes.
 - Les carpetes en xarxa només són pels tutors, per tant, en els 2 ordinadors dels alumnes no hi ha la configuració (i per aquest motiu es dedueix que no pertanyen al grup de xarxa "AJEM").
 - Tots els ordinadors dels tutors (6) haurien de tenir en funcionament la corresponent carpeta de xarxa. Però, efectivament, només ens uns quants està operatiu. Gairebé cap professional usa la carpeta en xarxa.
- Es confirma la varietat de programari (des del SO a resta de programari) i maquinari que existeix en els diversos ordinadors.

Adicionalment, en la fase de planificació s'havia establert com a ordinador de prova el de l'aula Eclipsi donat que es preveia que no s'usaria. Tanmateix, això ha canviat i sí que s'està utilitzant per un professional. Així, es descarta com a ordinador de prova i cal cercar una alternativa.

A partir d'aquestes definicions, en l'apartat següent, s'estableixen les especificacions pel desenvolupament i proves.

4.2.- Especificacions de desenvolupament i proves

A continuació es descriuen les especificacions de desenvolupament que són comunes pels dos serveis (Bloc Educatiu i Sistema de Gestió de la Informació):

- SGBD. Es defineix com a programa SGBD oficial del projecte **MySQL Community Edition 6.1** [22]. Les bases de dades dissenyades es lliuraran en format .sql.
- Diagrames i figures. Els diversos diagrames generats en la documentació no es lliuraran de manera separada, sinó integrats en els arxius de documentació. Es defineixen tres programes específics per crear diagrames: **Lucidchart** [25], **Dia** [27] i **Gimp** [18] i es permet utilitzar les eines de generació de figures dels processadors de textos i de MySQL.
- Programació de codi. Per aquest projecte, es defineix Java com a llenguatge principal de programació. Així, s'utilitzarà **HTML/JSP** (fins la versió 5), **Java 8 SE** i **SQL**. El primer s'usarà per programar les diverses interfícies web, el segon per programar els *Servlets* i el tercer per programar les bases de dades. S'utilitzarà l'IDE de desenvolupament **Netbeans 8.0** [20]. En aquesta fase del projecte i donats els recursos disponibles, no s'estableix cap compliment de qualitat de llenguatge HTML o altre. Es recomana, tanmateix, que les aplicacions generades siguin compatibles en **Firefox** (versió PC o tauleta) i **Chrome** (PC).
 - Disseny gràfic. S'estableix un disseny sobri per totes les interfícies a programar on l'únic element gràfic destacable sigui el logotip d'AJEM. Les tipografies a usar seran Arial o Slapa (font amb llicència lliure) de color majoritàriament negra o gris fosc. Pels colors de fons s'usaran grisos clars o colors pastel. Per la interfície web, per facilitar la

- programació d'elements gràfics i funcionals, s'usaran els paquets **Bootstrap** [28] i **jQuery** [29]. Es prioritzarà la programació gràfica mitjançant CSS.
- o Seguretat. Les claus d'usuari s'emmagatzemaran a la base de dades xifrades segons el protocol SHA-256.

Desplegament web. Com a servidor de les diverses pàgines web que constituiran les interfícies de gestor, tenint en compte que Java és el llenguatge oficial del projecte, s'usarà **Glassfish Open Source Edition 4.0** [30]. El principal motiu és que correspon a un dels servidor que més suporta Java.

A continuació, es descriuen les especificitats per cada un dels dos serveis.

4.2.1.- Bloc Educatiu

Com s'ha definit anteriorment, el Bloc es programarà amb Java (tant per la vista: pàgines .jsp com pel model: pàgines .java) utilitzant la IDE *Netbeans*. S'utilitzaran dues tècniques principals: els *servlets* i AJAX.

Una de les grans avantatges de la tecnologia AJAX és que és capaç de llegir informació per una pàgina web sense tornar-la a carregar. Així, per carregar els missatges de la base de dades quan hom es connecti a la pàgina web es realitzarà mitjançant una funció AJAX tipus `$(document).ready(function() {});`. El model de negoci de la crida AJAX que carrega els missatges es realitzarà mitjançant un arxiu de Java (**microblog.java**).

La resta del bloc funcionarà a través de *servlets* com es descriu a continuació. Aquests seran cridats per formularis tots ells usant el mètode *post*.

El bloc s'executarà mitjançant la pàgina principal **index.jsp**. Aquesta pàgina carregarà i mostrarà les darreres entrades del bloc (mitjançant la funció AJAX esmentada). També incorporarà també el formulari d'entrada de nous missatges i enllaços a tres serveis: filtrat de missatges, gestió d'usuaris i aules i eliminació de missatges.

Per escriure una nova entrada, caldrà introduir el missatge, el nom d'usuari i una fotografia o vàries alhora (opcional) en el formulari que serà tipus *multipart/form-data* (per suportar càrrega de fitxers). El formulari també contindrà un *checkbox* per indicar si l'usuari vol introduir títols a les fotografies. La data, el programa la crearà automàticament. Quan es premi el botó, s'accedirà al *servlet* **gestio.java** que validarà l'usuari, carregarà i tractarà la imatge (o imatges) si s'ha adjuntat (la convertirà en .jpg, li canviarà la mida fins a una amplada de 400 píxels mantenint la relació d'aspecte i al nom que porti li afegirà el *timestamp* per fer-la única en quan al mateix). En cas que l'usuari hagi especificat de titular les fotografies, el *servlet* cridarà una pàgina *jsp* on es visualitzaran les imatges i l'usuari podrà escriure'n els títol. En acabar, el botó del formulari retornarà a l'anterior *servlet*. Aquest, a continuació, desarà a la BD tota la informació text, incloent la ruta de la imatge modificada (o imatges). Les imatges (l'original i la modificada) seran desades en un directori creat per això. Finalment, el *servlet* retornarà a la pàgina inicial la qual, mitjançant la funció AJAX, refrescarà els missatges i ja s'inclourà el darrer creat.

En clicar el botó d'un dels altres tres serveis, sempre s'executarà el *servlet* **prefiltra.java**. Aquest re-direccionarà segons el servei seleccionat. Si és el de gestió o filtrat de missatges, en el mateix *servlet*,

primer s'accedirà a la BD per carregar el nom de les aules, que es traspassarà a la pàgina *.jsp* que es cridarà mitjançant una variable tipus llista.

Si s'utilitza el servei de filtre, el *servlet* direccionarà a la pàgina **filtre.jsp**. Aquesta pàgina permetrà que l'usuari seleccioni si es volen filtrar els missatges per aula, calendari o les dues coses. Pel cas de l'aula existirà un desplegable amb les aules existents (es carregarà mitjançant ordres *jstl* i a partir de la variable rebuda de **prefiltra.java**). Pel cas de calendari, es mostrarà un calendari gràfic. Amb la tria efectuada, el botó del formulari executarà el *servlet* **filtre.java** que carregarà la pàgina inicial però enviant-li els criteris de filtratge, els quals seran carregats per la funció AJAX d'obtenció de missatges. Aquests criteris seran rebuts per **microblog.java** i segons quins siguin, realitzarà una cerca o una altra a la SGBD.

Si s'utilitza el servei de gestió, el *servlet* direccionarà a la pàgina **gestiona.jsp**. Aquesta pàgina contindrà formularis per gestionar tant els usuaris com les aules. Les aules es podran crear o eliminar. Tanmateix, mai es podran eliminar aules les quals tinguin missatges, primer, caldria eliminar els darrers (no es permeten missatges orfes). Els usuaris es podran crear, eliminar i visualitzar. Si es creen, caldrà indicar si són administradors o no i a quina aula corresponen. Sempre es podrà eliminar un usuari (al fer-ho, es visualitzarà un avís). Pel cas de l'aula, hi haurà un desplegable amb les aules que existeixen (es carregarà mitjançant ordres *jstl* i a partir de la variable rebuda de **prefiltra.java**). En tots els casos, també caldrà indicar el nom de l'usuari que vol fer les modificacions, el qual serà validat pel *servlet* (haurà de ser administrador). Un cop es premi el botó, actuaran el *servlets* **aulas.java** o **usuaris.java** segons escaigui. Aquests *servlets* realitzaran les funcions indicades i retornaran a la pàgina principal.

Finalment, si s'utilitza el servei d'eliminar missatges, el *servlet* direccionarà a la pàgina **elimina.jsp**. Aquesta pàgina inclourà un formulari on caldrà indicar-hi el nom de l'usuari que vol executar l'acció i el número de missatge. Quan es premi el botó, el *servlet* **elimina.java** gestionarà l'operació. Comprovarà que l'usuari sigui l'autor o un administrador i si ho és, eliminarà el missatge. Prèviament a l'eliminació del missatge, s'eliminaran de la BD les fotografies associades (les rutes, però no els arxius). Finalment, es retornarà a la pàgina inicial.

4.2.2.- Sistema de Gestió de la Informació

Com s'ha definit anteriorment, el Gestor es programarà amb Java (tant per la vista: pàgines *.jsp* amb etiquetes *jstl* com pel model: pàgines *.java*) utilitzant la IDE *Netbeans*. S'utilitzaran *servlets* com a tècnica principal, cridats per formularis que usaran el mètode *post*. Per alguns casos, també s'utilitzarà AJAX.

El bloc s'executarà mitjançant la pàgina principal **index.jsp**. Aquesta servirà únicament per validar un nom d'usuari i contrasenya i en cas d'èxit, re-direccionar la pàgina principal del Gestor (únicament accessible a través d'aquest procediment, com la resta de pàgines que conformen el Gestor). L'accés ja serà desat en la BD i complementat per les accions que efectui a continuació.

La pàgina principal del gestor contindrà tres seccions diferenciades: la de la gestió del CEE, la de la gestió d'aula i la del gestor fotogràfic. En funció del rol de l'usuari validat, es tindrà accés a una o més seccions. Addicionalment, per facilitar l'operativa en el gestor, donat que l'atribut "persona" és clarament el clau i que aquest ve identificar inequívocament pel NIF, hi haurà una eina de cerca de

persones per camp de text lliure (per exemple, es podran cercar persones a partir de la introducció d'un cognom, del nom, etc.). Es retornarà una llista amb les persones coincidents, de les quals s'informarà el NIF, nom i cognoms.

A partir d'aquí, cada secció inclourà les seves opcions d'ítems a gestionar segons s'ha definit anteriorment. Per cada ítem, es podran realitzar les següents accions: afegir, modificar, eliminar o visualitzar. Com s'ha indicat, tot el Gestor funcionarà a través de formularis *post* i *servlets*.

Donat que el Gestor té ja certa complexitat (a diferència del Bloc), la programació seguirà una estructura de Model – Vista – Controlador. Així, l'estructura del projecte serà la següent:

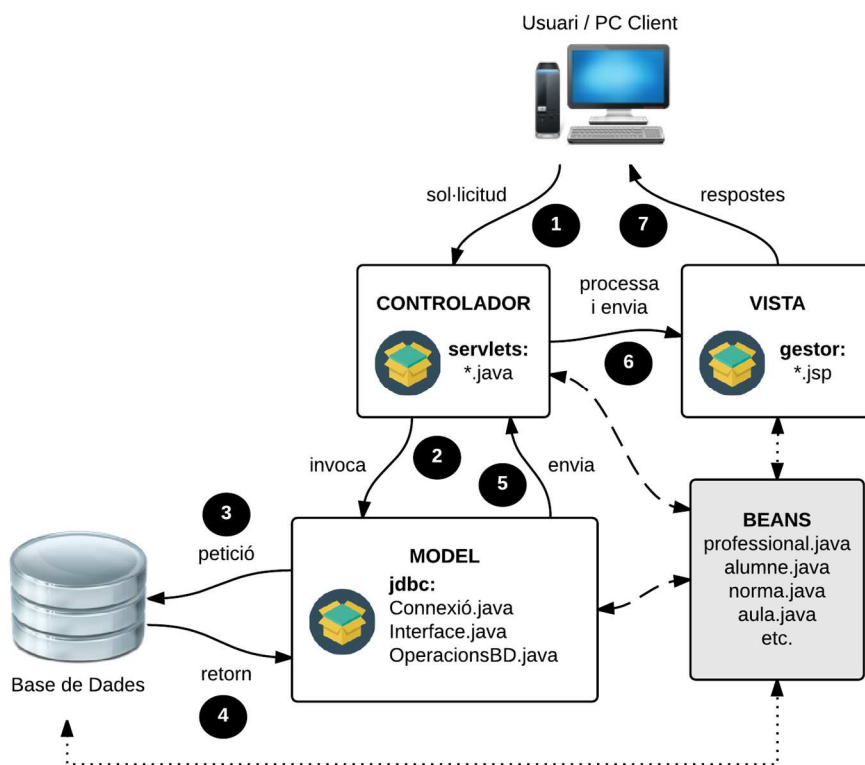


Figura 21. Esquema Model-Vista-Controlador del Sistema de Gestió de la Informació

El procés s'inicia quan un usuari des d'un PC client realitza una petició (sol·licitud). Aquesta és gestionada pel controlador (*servlet*) el qual, si és necessari (que ho és en la majoria dels casos), requereix l'ús del SGBD. Aquest està format pel conjunt de peticions que gestionen les dades de la BD (model). El model està estructurat en tres arxius: l'*Interface*, que actua d'arxiu índex de totes les operacions de gestió de dades, l'*OperacionsBD* que implementa les instruccions de l'*Interface* i per tant, conté totes les subrutines d'operacions a la BD i el *Connexió* que conté les dades de connexió a la BD. Així, mitjançant el model es realitza la consulta a la BD i s'obté el retorn. El model el retorna al controlador que, a través dels arxius *jsp* de vista, subministra la resposta a l'usuari que ha realitzat la petició.

Tot el procés usarà els *beans*, que són els objectes específics creats per aquest cas i que corresponen, majoritàriament, als diversos recursos identificats.

Finalment, comentar que les subrutines en les quals estant programades les consultes a la BD usaran la tècnica *PREPARED STATEMENT*, en les quals, es defineix primerament totes les consultes amb símbols *?*, que en les subrutines són reemplaçats pels valors corresponents.

4.2.3.- Programari de base (T3)

Donats els aspectes definits anteriorment i els especificats en la planificació, el proposa el següent disseny de la implementació és la següent:

1. Establir com a ordinadors de proves els 2 ordinadors dels alumnes (nº 4 i nº 6)
 - Són els menys crítics: no contenen dades rellevants, no s'usen pels professionals del centre, no usen carpetes de xarxa.
 - Instal·lació del SO Linux Mint i resta de programa planificat.
2. Còpia de dades d'usuari
 - Després d'una reunió amb els professionals del CEE, se'ls ha comentat que tenen un gran volum de dades emmagatzemades en els diversos ordinadors i de forma només local. Els professionals van comentar que són magatzems històrics incontrolats (el material que hi ha no està inventariat). Se'ls ha proposat que aquesta seria una bona ocasió per inventariar-lo i eliminar allò que no calgui guardar per tal d'higienitzar el magatzem i racionalitzar l'ús d'espai de disc. En aquests moments, cal que els professionals prenguin una decisió i en funció de la mateixa, les actuacions a realitzar i la programació en quan a calendari variarà.
 - i. Si decideixen efectuar l'inventari, necessitaran temps per fer-ho que s'haurà de disposar. I d'altre banda, això pot fer possible una sensible disminució de les dades a guardar, fet que facilitarà el procés de guardat de dades i procedir amb el planificat inicialment: donar nou format als discs durs i instal·lació de Linux Mint.
 - ii. Si decideixen no efectuar l'inventari, es podran iniciar actuacions pròximament. Caldrà comprovar la viabilitat de traslladar les dades de tots els ordinadors. Si és viable, s'efectuarà el mateix procediment que en l'apartat anterior, i si no, es procedirà a la segona opció planificada que és mantenir el sistema operatiu actual de l'ordinador amb les dades d'usuari que conté, i que la instal·lació de nou programari afectés només a la resta de programari.
3. Primera etapa de formació als professionals.
 - Usant els dos ordinadors enllestits, es farà una primera formació bàsica sobre ús del SO i resta de programari (sobretot, processador de textos).
4. Segona actuació d'instal·lació de programari de base a la resta d'ordinadors. Aquesta actuació està supeditada a la resolució de l'ítem nº 2.
5. Segona etapa de formació als professionals del CEE
 - Nou programari de base
 - Nous serveis: Bloc Educatiu i Sistema de Gestió de la Informació

4.3.- Requisits d'implantació

A continuació es defineixen els requisits d'implantació, els quals es realitzen sota dos enfocaments, el de l'usuari del subsistema i el tecnològic i els seus recursos.

Des del punt de vista de l'**usuari** cal tenir en comptes dos grans aspectes: la persona que mínimament administrerà tots els nous sistemes i el procés de canvi del programari i la implantació dels dos nous serveis (Bloc Educatiu i Sistema de Gestió de la Informació), que afecta a tots els usuaris.

En relació a l'administrador, donat els coneixements dels professionals del CEE i la seva dedicació, s'ha dissenyat tot el projecte de manera que sigui el més senzill possible i requereixi la menor atenció per part de personal. En aquest sentit i per cada una de les tasques s'estableixen les següents consideracions:

- **Nou programari de base (T3).** Per aquesta tasca i a efectes d'administrador, com estava planificat, no es defineix cap requisit. El servei d'administrador que es pugui requerir eventualment es gestionarà de la mateixa manera que es gestiona actualment: usant serveis externs. El nou programari no ha d'afegir més complexitat al sistema ni més casos de requeriments de serveis externs que amb el programari actual. Al contrari, es preveu que el sistema sigui més robust i per tant, que es tinguin menys incidències.
- **Bloc Educatiu (T2.1).** Per aquest servei i a efectes d'administrador, com també estava planificat, no es requereix el nomenament d'un administrador tècnic com a tal. El fet que el servei estigui implementat en un servidor extern gestionat per personal extern, implica que tots els serveis d'administració vagin a càrrec d'aquest i per tant, no calgui intervenció de personal del CEE. En tot cas, sí que cal definir una persona d'enllaç entre el CEE i el personal del servei extern per possibles necessitats molt eventuals: recuperació d'alguna imatge, re-definició del servei, etc. S'estableix que el/la director/a del CEE es faci càrrec d'aquesta tasca.
- **Sistema de Gestió de la Informació (T2.2).** Aquest servei també s'ha dissenyat minimitzant la necessitat d'un administrador, però donat que és un servei totalment intern i encara que requereixi accions molt puntuals, cal definir un administrador. Les tasques que hauria de realitzar són:
 - Conèixer el funcionament del subsistema, a la documentació del qual tindrà accés.
 - Comprovar i monitorar el correcte funcionament del subsistema.
 - Conèixer els possibles riscos i com es poden solucionar i incorporar aquestes actuacions dins de la política de còpies de seguretat de l'empresa.

Aquestes tasques són molt coincidents amb les que ja efectua el Gerent d'AJEM en tant en quan és l'administrador intern de l'empresa del servei de còpies de seguretat que hi ha implementat. Per tant, és la persona més adequada per assumir les tasques. Addicionalment, les còpies de seguretat de la BD estaran automatitzades i integrades dins l'actual sistema i metodologia de còpies de seguretat.

En relació al procés de canvi a què es veuran implicats tot el personal del CEE (implantació del nou programari de base i posada en marxa dels nous serveis d'informació) es dissenyen dues accions:

- Elaboració de guies/manuals d'usuari.
- Formació a tots els professionals. La formació es farà en dues etapes: una primera en relació al nou programari de base i abans que s'iniciï la implantació del mateix. I una segona després de la implantació, en la qual es tractarà aspectes del nou programari (consolidació de coneixements) i també dels nous serveis: Bloc Educatiu i Sistema de Gestió de la Informació.

Des del punt de vista **tecnològic**, es defineixen diversos requeriments per cadascuna de les tres tasques:

- **Nou programari de base (T3).**
 - Salvaguardar totes les dades emmagatzemades localment en cadascun dels ordinadors (veure anàlisi de l'apartat anterior).
 - Programar les instal·lacions fora de l'horari habitual de classes.
 - Realitzar les intervencions ordenadament, començant per les dues màquines d'ús majoritari dels alumnes.
- **Bloc Educatiu (T2.1).** Aquesta tasca no té cap requisit especial, ja que s'implanta en un servidor extern que no interfereix en res al CEE i el manteniment va completament a càrrec pel personal del servei extern.
- **Sistema de Gestió de la Informació (T2.2).**
 - Abans d'implementar el servei, el servidor s'haurà d'adequar en quan a programari: instal·lació del servidor web i de la SGBD.
 - Aquesta tasca d'adequació s'haurà de realitzar en un moment de mínima interferència:
 - Fora d'horari habitual de treball.
 - Que no correspongui a cap moment d'execució de còpies programades.
 - La base de dades de projectes haurà de ser accessible pel servei que fa les còpies de seguretat, i els canvis en la seva forma d'accés, usuaris i contrasenyes hauran de ser comunicats al responsable del servei per a evitar interrompre la prestació del servei.
 - Prèviament a cada procés de còpia, es comprovaran les condicions inicials del servei, com ara:
 - Prevenció per evitar que modificacions simultànies al procés de còpia provoquin un error en el servei.
 - Dimensionament de recursos per assegurar que el procés de còpia comença i acaba correctament.
 - Integració del procés de còpia de la base de dades amb la resta de subprocessos de còpia.

5.- Desenvolupament

5.1.- Planificació de les activitats d'integració de sistema

La figura 1 mostra la planificació de l'etapa de desenvolupament (ampliada a l'annex 1). Aquesta etapa afecte sobretot a la generació dels dos serveis: Bloc Educatiu (T2.1) i Sistema de Gestió de la Informació, que inclou el gestor fotogràfic (T2.2 i T2.3). En la tasca d'implementació del nou programari de base no hi ha desenvolupament a banda de la generació del material didàctic/formatiu (igual que per T2.1 i T2.2) que també estant incloses en el cronograma (T4). Les tasques de correcció s'han inclòs en la fase de la implementació. I finalment, les tasques generals de gestió del projecte, elaboració d'informes del TFM, etc., tampoc s'han inclòs (ja cobertes en la planificació general efectuada).

La planificació (tant cronograma com dedicació) s'ha efectuat a partir de la general, efectuada en la planificació del projecte. A l'entrar en detall, degut a les activitats ja realitzades fins ara i l'estat avançat del projecte, existeixen petites adaptacions i canvis.

Així, la planificació de desenvolupament es realitza principalment sobre el Bloc Educatiu i el Sistema de Gestió de la Informació. Donat que la magnitud del dos serveis és ben diferent, l'estratègia de planificació també ho és. El punt en comú pels dos casos és que primer s'aborda el disseny i desenvolupament de la base de dades i després el desenvolupament de codi de l'aplicació.

Pel cas del Bloc Educatiu, s'han classificat i distribuït les tasques per objecte de gestió: gestor d'entrada de missatges, gestor d'usuaris, gestor de visualització de missatges, etc. Les tasques es desenvolupen bastant en paral·lel (són tasques curtes) i el procés finalitza amb la primera versió completa disponible per iniciar la fase de proves.

Pel cas del Gestor, les tasques s'han agrupat per funcionalitat: desenvolupament de tots el paquets de gestió principal dels objectes (professional, alumne, incidència, etc.), paquets de modificació d'objectes, etc. I cadascun d'aquests paquets conté diversos arxius, un per cada objecte. La justificació és que l'estructura de cada arxiu per cada funció és molt similar per tots els objectes, encara que cadascun té condicionants diferents. D'acord amb el volum d'objecte i tal i com es va dissenyar, el desenvolupament del Gestor segueix una rigorosa estructura Model – Vista – Controlador. Això també es té en compte en la divisió de tasques, existint tasques dedicades exclusivament a la generació dels *beans* o les operacions a la BD. De fet, l'arxiu que conté totes les accions sobre la BD és dels més extensos del projecte. Donat el volum de treball de cada paquet, les tasques es realitzen més en sèrie. Com en el cas anterior, el procés finalitza amb la primera versió completa disponible per iniciar la fase de proves.

D'acord amb la planificació inicial, primer es desenvolupa el Bloc i després el Gestor, tot i que hi ha cert solapament.

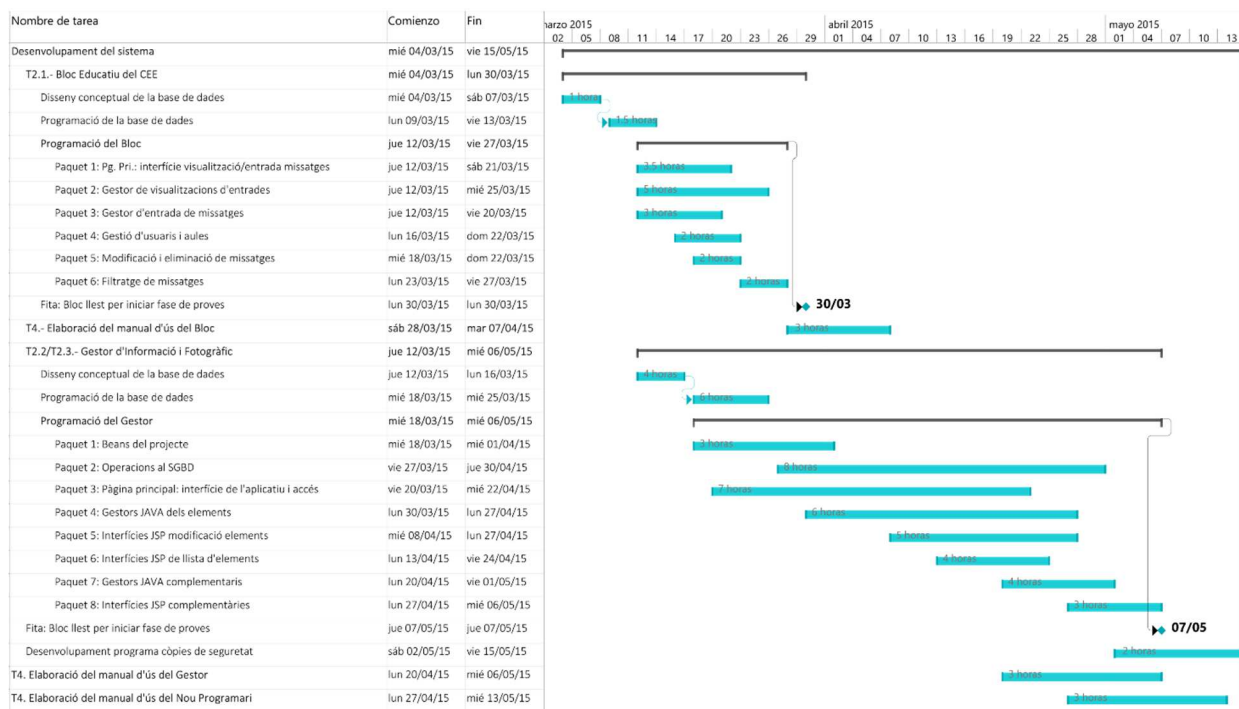


Figura 22. Cronograma de la fase de desenvolupament

En relació al Sistema de Gestió de la Informació, els paquets 7/8 referents a gestors Java i interfícies JSP complementàries fan referència a opcions com canvi de clau o fotografia de l'usuari, descàrrega de l'arxiu d'accessos/accions, tancament de sessió, etc.

El cronograma també recull el desenvolupament d'un programa en Java per gestionar les còpies de seguretat de les dades del Gestor al servidor del CEE. El programa, permetrà programar i executar *backups* de dades de la BD i arxius a carpetes sobre les quals el sistema actual realitza còpies de seguretat.

Finalment, comentar que en el diagrama de *Gantt* no s'han inclòs els recursos humans ja que tot ho realitza una única persona (l'autor). Totes les hores de dedicació indicades en les barres són a càrrec del mateix.

En aquest projecte, el seguiment de la planificació s'ha efectuat mitjançant dos mecanismes: (1) correspondència amb el tutor de la UOC, en la qual s'ha anat explicant l'evolució detallada de les tasques, les activitats dutes a terme en el darrer període i les activitats previstes pel següent i (2) mitjançant reunions amb els professionals del CEE (incloent el tutor del centre) amb els mateixos continguts. La documentació generada al respecte són les correspondències i les actes de les reunions (s'adjunten en l'annex 2). Aquest doble seguiment de les activitats per part dels dos tutors i cadascun sota les seves perspectives han assegurat un desenvolupament adequat.

5.2.- Elecció de llicències

L'elecció de les llicències s'ha efectuat en consens entre la direcció del CEE i l'autor. A continuació s'identifiquen i es justifiquen les tries efectuades, en funció de cadascun dels productes desenvolupats i/o usats.

5.2.1.- Programes desenvolupats: Bloc educatiu i Gestor de la Informació

Ambdós codis han estat programats en **Java** i usant la **IDE Netbeans 8.0** [20]. La majoria del codi és de producció pròpia, malgrat s'han usat algunes llibreries externes:

- [Bootstrap](#) [28]. Paquet usat per aspectes de format. Llibreria distribuïda sota [Llicència MIT](#) [31].
- [jQuery](#) [29]. Paquet usat per complements gràfics com l'ús d'un calendari per seleccionar data o control i visualització dels caràcters en camps de text. Llibreria distribuïda sota [Llicència MIT](#).
- [Json](#) [32]. Format usat per transferir dades entre les subrutines de les aplicacions que usen [AJAX](#). [Llicència pròpia](#).
- [Fileupload](#) [33]. Component usat per carregar arxius en formularis web tipus *multipart/form-data*. [Llicència Apache v2](#) [34].
- [JExcelAPI](#) [35]. Component utilitzat per generar l'arxiu en format *Microsoft® Excel®* que conté l'historial d'accessos i accions realitzades al Gestor. [Llicència Lesser General Public License version 2.0](#) (LGPLv2) [36].

Tant les llicències *MIT*, *Apache* com la pròpia de *Json* permeten usar el codi per desenvolupar programes amb gairebé qualsevol llicència, inclosa la comercial. Malgrat que totes requereixen citar-ne el copyright. La que cal tenir més en compte és la *LGPL* que no permet produir programes amb qualsevol llicència. Per tant, en funció del què es desitgi, pot no ser vàlida. Tanmateix, es fa servir en una acció molt puntual i per tant, seria molt fàcilment prescindible (per exemple, generant un simple arxiu de text per descarregar la llista d'accessos, la qual cosa es realitzarà molt puntualment).

Tenint en compte l'anterior, s'opta per elegir una llicència que, d'una banda, mantingui la informació de copyright, i d'altra banda, sigui el més permissiva possible. És a dir, que permeti en el futur qualsevol opció (inclosa l'opció de convertir el codi en propietari). Això respon al concepte de llicència permissiva, tipus la *BSD* [37]. I per tant, és l'elegida (tenint en compte l'aspecte puntual de la llibreria *JExcelAPI*).

Cal considerar que la finalitat del CEE no és produir ni vendre programari, ni és una entitat de lucre, ans el contrari. És un centre educatiu amb vocació i servei públic. Per tant, sota aquest precepte, la idea original és facilitar el codi obert i la seva divulgació. Tanmateix, cal considerar també que el CEE requereix de finançament per poder desenvolupar la seva activitat. Les donacions o mecenatge són escasses i el finançament públic està en clar descens. Per tant, es requereixen també altres fonts de finançament, i per aquest motiu, no es descarta cap possibilitat també en aquest cas.

Esquema resum de la llicència BSD [37]

Copyright © *el propietari*. Tots els drets reservats.

Es permet la redistribució en font i en binari, amb modificació o sense, sempre que es compleixin les condicions següents:

- 1) Les redistribucions en font han de retenir la nota de *copyright* i fer una llista d'aquestes condicions i la limitació de garantia.
- 2) Les redistribucions en binari han de reproduir la nota de *copyright* i fer una llista d'aquestes condicions i la limitació de garantia en la documentació.
- 3) Ni el nom del *propietari* ni el dels que hi han contribuït no poden utilitzar-se sense permís per a promocionar productes derivats d'aquest programa.

Aquest programa es proporciona "tal qual", sense garanties expresses ni implícites, com ara la seva aplicabilitat comercial o la seva adequació per a un propòsit determinat. En cap cas el *propietari* serà responsable de cap dany causat pel seu ús (inclosa la pèrdua de dades, la pèrdua de beneficis o la interrupció de negoci).

5.2.2.- Nou programari de base instal·lat als ordinadors del CEE

En aquest cas, no hi ha hagut cap tipus de desenvolupament per part de l'autor, i els diversos programes s'han usat tal qual. Per tant, cadascun d'ells s'usa sota la seva respectiva llicència original. A continuació es mostra una taula amb els principals programaris implementats i la seva llicència.

Programari	Llicència
<i>Linux Mint</i>	Llicència GNU-GPL. I la majoria de components de tercers són també lliures.
<i>LibreOffice</i>	Llicència pública general menor del projecte GNU (LGPL v3)
<i>Gimp</i>	Llicència GNU-GPL.
<i>Audacity</i>	Llicència GNU-GPL.

5.2.3.- Guies d'ús desenvolupades i document de Política d'IT del CEE

Per aquests casos, s'aplica el mateix criteri detallat en l'apartat 1.2.1, en relació al codi generat, però usant llicència de *Creative Commons* [38], potser més especialitzada i usada per aquest tipus de producte. I per tant, tot el material generat se li assigna, concretament, la llicència de Reconeixement i Compartir Igual (CC BY-SA). Aquesta llicència comparteix les mateixes bases que la llicència tipus BSD seleccionada i justifica pel codi.



Aquest apartat també inclou el propi document de política interna d'IT del CEE. I en aquest cas, d'una forma més sensible: el model implementat en aquest centre pretén també servir d'exemple per d'altres. La difusió d'aquest document, doncs, es pretén que contribueixi a aquest objectiu.

5.2.4.- Documents que conformen el TFM i Pràctiques Externes (lliurables de la UOC)

Per aquest tipus de documents, la tria de llicència es realitza seguint les recomanacions de la Normativa per a la presentació del Projecte Final de Màster en Programari Lliure de la UOC. Aquesta, recomana l'ús d'una llicència lliure com per exemple GFDL o CC-BY-SA. De fet, la darrera recomanada ja és la que s'ha elegit per la resta de documentació, i, per tant, és l'elegida (contribuint a la uniformització de llicències).

Finalment, mencionar que les dades que s'usen en els diversos programes, s'exclouen d'aquest apartat, ja que s'entén que no formen part del projecte en sí. En qualsevol cas, són dades de propietat i gestió del CEE, sotmeses a la Llei de Protecció de Dades.

5.3.- Entorn de desenvolupament

Com s'havia planificat i dissenyat, el desenvolupament del codi d'aquest projecte s'ha efectuat utilitzant classes **Java** [39] pel model de negoci i una combinació d'**HTML** [40], **Java Web (JSP, Javascript i JSTL** [41]) i **CSS** [40] pels estils.

En alguna ocasió (càrrega de missatges en el Bloc o descàrrega de l'arxiu d'accessos en el Gestor) el model de negoci es realitza mitjançant funcions AJAX que criden classes de Java i aquestes retornen informació en format JSON. Tanmateix, la gran majoria de model de negoci en les dues aplicacions es gestiona mitjançant *servlets*. Aquestes s'implementen en classes Java utilitzant l'etiqueta *@WebServlet*.

Tots els *servlets*, es gestionen sota mètodes *ProcessRequest* i *POST*, bescanviant informació i redirigint amb instruccions *setAttribute* i *forward*. Així, l'estructura general de tots els *servlets* és:

```
@WebServlet("/Nom_servlet")
public class GestorInformes extends HttpServlet {
    protected void processRequest(HttpServletRequest request, HttpServletResponse
    response) throws Excepcions {
    }
    @Override
    protected void doPost(HttpServletRequest request, HttpServletResponse response)
    throws Excepcions {
        request.setAttribute("nom_variable", valor_variable);
        getServletContext().getRequestDispatcher("URL_intern").forward(request,
        response);
    }
}
```

Per les interfícies d'usuari, totes les pàgines són JSP i combinen codi HTML, Javascripts i codi JSTL. El **Javascripts** s'utilitzen per executar funcions puntuals com desplegar o amagar formularis, utilitzar l'ús del calendari gràfic per seleccionar una data o gestionar les funcions AJAX. El codi **JSTL** s'utilitza per gestionar la informació provinent dels *servlets*, ja sigui per llegir-la, per tractar-la sota condicions IF o per iterar amb ella amb blocs FOR. Per exemple, resulta d'utilitat la combinació de bloc FOR i IF en la visualització de desplegables en formularis on cal que un dels valors predeterminats segons informació de la base de dades estigui seleccionat.

L'estil de les diverses interfícies es gestiona sobretot amb les llibreries *Bootstrap* [28] i *jQuery* [29]. Tanmateix, existeixen personalitzacions gestionades, en bona mesura, en un arxiu d'estils CSS.

Cal considerar que el codi que s'ha generat, donat el condicionant temporal del projecte, no està optimitzat. L'objectiu era programar tota la funcionalitat i sense errors. Aspectes que caldria millorar són evitar algunes instruccions d'estil que s'han inclòs en pàgines JSP, revisar els noms de mètodes Java en relació a l'ús de majúscules i minúscules, revisar les funcions *catch()*, etc. En resum, adequar-se completament als llibres d'estil de cadascun del codis (HTML, Java, etc.)

En relació a l'entorn de desenvolupament tot el codi s'ha programat en l'IDE **Netbeans 8.0** [20]. Aquesta IDE ofereix una plataforma totalment suficient per realitzar totes les tasques necessàries, tenint en compte l'addició de llibreries i *plugins*. Netbeans, tot i que permet treballar amb diversos llenguatges (C++, PHP, etc.), va ser inicialment desenvolupat i optimitzat per programar en Java. Per aquest projecte, s'ha treballat amb el servidor **Glassfish Server Open Source Edition v 4.0** [30], que el propi Netbeans porta incorporat per defecte. Aquest servidor és el mateix (model i versió) que s'usarà en la implementació de les dues aplicacions.

El gestor de base de dades usat en tot el projecte (tant en el desenvolupament com en la implementació) ha estat **MySQL Community Server (GPL) v. 5.6.19** [22]. La interconnexió entre Java, MySQL i Glassfish està perfectament resolta mitjançant controladors madurs. S'han utilitzat també les eines de MySQL incorporades a **MySQL Workbench 6.1** per generar diagrames gràfics dels diversos esquemes de base de dades.

A continuació, es realitza una descripció funcional i organitzava del codi desenvolupat per les dues aplicacions (Bloc Educatiu i Sistema de Gestió de la Informació). El Bloc Educatiu està dissenyat perquè sigui usat com a material didàctic pels professionals i usuaris del CEE i no com a projecció exterior del mateix. Les bases de disseny principals són la senzillesa i limitar-ne l'ús als propis professionals i usuaris. El Sistema de Gestió de la Informació, en aquesta primera versió, està dissenyat per esdevenir un repositori de dades del CEE i facilitar l'accés a la informació. Per tant, no té funcions de planificador, no realitza accions actives.

5.3.1.- Bloc educatiu

A continuació es descriu la funcionalitat del Bloc Educatiu, segons el disseny efectuat i codi generat.



Figura 23. Imatge d'una interfície del Bloc Educatiu.

- **Visualització de missatges.** A l'accedir al bloc, es mostren els darrers 20 missatges introduïts.
 - **Filtrat.** Tanmateix, l'eina permet filtrat els missatges a visualitzar segons 3 criteris: filtrat per aules, filtrat per calendari o filtrat per ambdós paràmetres.
- **Introducció de missatges.** Els missatges s'introdueixen en la pàgina principal del bloc i fent clic al rètol "introdueix un missatge", la qual cosa desplega les opcions. Un missatge s'introdueix a partir d'un nom d'usuari (el qual està associat al nom d'una aula), el cos del missatge, una data (opcional) i una o diverses fotografies de les quals s'hi pot afegir un títol o no.
- **Eliminació o modificació de missatges.** En la pàgina principal on es visualitzen els missatges, cadascun està també identificat per un número. Si es fa clic damunt del número, s'accedeix a la pàgina d'eliminació o modificació del missatge. La modificació permet canvis en el cos del missatge o la data.
- **Gestió d'usuaris i aules.** El bloc també conté una pàgina per gestionar aquests dos paràmetres. Sobre l'usuari, que es pot afegir o eliminar, si s'afegeix, se li assigna nom d'usuari, una descripció, quina aula representa i si és administrador o no. Ser administrador implica poder gestionar altres usuaris i aules i gestionar els missatges de tots (sinó, cada usuari només pot gestionar el seus). En relació a les aules, es poden crear o eliminar noms d'aules.

Cal referir-se al capítol de disseny del sistema d'aquest TFM per ampliar aquest tipus d'informació.

La figura 24 mostra com s'ha desenvolupat el codi de l'aplicació a nivell organitzatiu, a partir de les diverses subrutines generades en Java i JSP (a l'informe de Pràctiques Externes s'adjunta tot el codi):

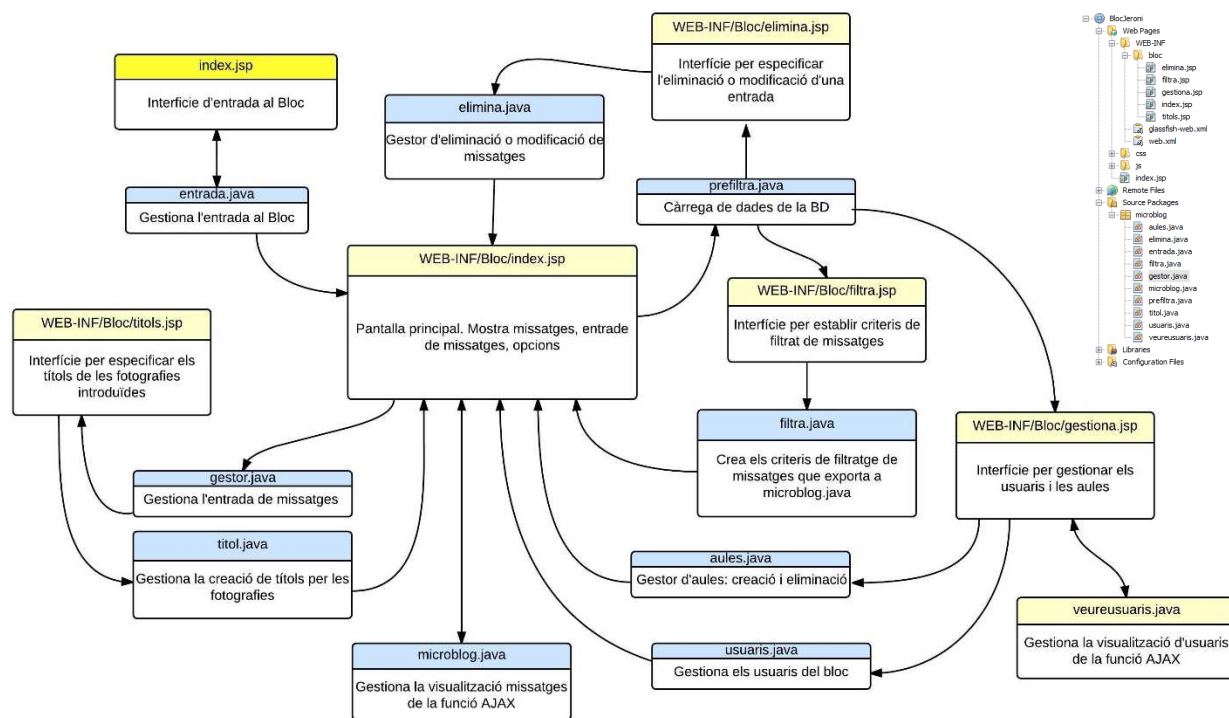


Figura 24. Estructura del codi del Bloc Educatiu

En relació al disseny gràfic del Bloc, s'ha realitzat un disseny sobri amb pocs ornaments gràfics. L'únic element gràfic és el logotip d'AJEM. A partir d'aquí, el protagonisme l'adquireixen les fotografies dels missatges. Els colors del bloc són tonalitats grises defugint del negre intens, més agressiu. Totes les pàgines tenen en comú una capçalera i un peu de pàgina fixos, i un cos de pàgina on hi ha la informació principal amb fons gris clar.

Execució de proves

El servei del Bloc Educatiu es subministra a partir d'un servidor extern del CEE. Un cop l'aplicació va estar desenvolupada i provada localment per l'autor, es va activar una versió de proves en el servidor definitiu (<http://blocjeroni.necessito.cat>). Tanmateix, per aquesta funció de proves, per accedir-hi calia introduir una clau d'accés. És a dir, en condicions de funcionament normal, tothom accedeix al bloc introduint l'adreça URL, però en l'etapa de proves, només usuaris amb permís a través d'una clau hi podien accedir.

L'execució de proves es va iniciar el 27/03/2015 i va finalitzar el 08/05/2015. Durant aquest temps, els professionals del CEE i d'altres actors van anar provant l'eina, detectant errors i millores a fer en quan a funcionalitat i usabilitat, les quals van ser totes considerades i aplicades. Un resum de les mateixes són (canvis o afegits respecte al disseny inicial):

- Possibilitat d'incorporar títols a les fotografies.
- Possibilitat d'incorporar dates a les entrades.
- Possibilitat de modificar entrades.
- Possibilitat de visualitzar llista d'usuaris.
- Obtenir un missatge d'avertència abans d'eliminar un usuari.
- Afegir una descripció de l'usuari en la seva creació.
- Diverses millores de presentació del Bloc (*layout*) i correcció d'errors.

5.3.2.- Sistema de Gestió de la Informació

El Sistema de Gestió de la Informació té una complexitat superior al Bloc. Aquest fet requereix de més definicions. Així, en primer lloc, es llisten els objectes que l'aplicació permet gestionar, juntament amb els seus atributs (estant el clau subratllat), d'acord amb el disseny que es va realitzar:

- *Professional*. NIE, nom, cognoms, tutor (si ho és o no), horari del professional (arxiu adjunt), clau d'accés, usuari actiu (si ho és o no)¹⁷, i nom de categoria.
- *Alumne*. NIF, nom, cognoms, data de naixement, nom del pare, nom de la mare, nom de la parella del pare, nom de la parella de la mare, nom del tutor, adreça, adreça electrònica, 4 camps de telèfon, número de la targeta sanitària, compartit (si l'alumne es comparteix amb un altre centre o no), any d'entrada al CEE, 4 camps de diagnòstic mèdic, 4 camps de medicació, programa (arxiu adjunt).

¹⁷ Si un professional esdevé baixa del CEE i si ha realitzat activitat al Gestor, no es podrà eliminar del sistema ja que l'històric d'activitat i dades ha de romandre. En conseqüència, per impedir-li l'accés i evitar que realitzi cap més acció, l'usuari amb rol de director o administrador pot marcar-lo com a usuari "no actiu".

- *Classes*. Les classes es defineixen inequívocament per la suma d'un nom aula i curs acadèmic (veure l'apartat de definicions generals, per exemple: Logopèdia, curs 2015/16). Ambdós ítems sempre es gestionen a partir de llistes que es poden modificar (per exemple, existeix una llista pre-definida de noms d'aula que es modifica). Cada classe conté: nom curs, llista de professionals tutors assignats, llista d'alumnes que l'integren i horari de la classe (arxiu adjunt).
- *Faltes d'assistència*. Alumne que ha faltat, data, motiu, professional tutor i número d'identificació de la falta.
- *Reunions*. Alumne sobre el qual s'ha fet la reunió, professionals del CEE que han assistit, altres assistents, data i hora, número de la reunió predecessora, acta o contingut de la reunió (arxiu adjunt) i número d'identificació de la reunió.
- *Informes*. Alumne del qual es realitza l'informe, professionals involucrats, data, títol, informe (arxiu adjunt) i número d'identificació de l'informe.
- *Incidències*. Nom de la incidència, data i hora, professional del CEE que la reporta, alumnes involucrats, informe de la incidència (arxiu adjunt) i número d'identificació de la incidència.
- *Normes*. Títol, text de la norma (arxiu adjunt) i número d'identificació de la norma.
- *Fotografia*. 4 etiquetes que defineixen la fotografia (substitueixen el títol i els camps són opcionals), aula a la que pertany, llista d'alumnes associats, data, fotografia (arxiu adjunt) i número d'identificació de la fotografia.



Figura 25. Imatge d'una interfície del Sistema de Gestió de la Informació.

Per tal de donar compliment a la Llei de Protecció de Dades, a banda dels anteriors ítems (propis de gestió), el sistema també emmagatzema els accessos a la base de dades, les accions realitzades, per qui i quan:

- *Accés*. Professional que accedeix al gestor, data i hora, acció que realitza i número d'identificació de la mateixa.

Aquestes accions estant tipificades segons mostra la següent taula:

Llistat d'accions disponible en l'accés al Sistema de Gestió de la Informació del CEE				
acces	afegir_professional	modificar_professional	eliminar_professional	llistar_professional
afegir_alumne	modificar_alumne	eliminar_alumne	llistar_alumne	afegir_nomaula
eliminar_nomaula	assigar_alumnes_aula	afegir_aula	modificar_aula	eliminar_aula
llistar_aula	afegir_norma	modificar_norma	eliminar_norma	llistar_norma
afegir_assist	eliminar_assist	llistar_assist	afegir_incidencia	eliminar_incidencia
llistar_incidencia	afegir_informes	modificar_informes	eliminar_informes	llistar_informes
afegir_reunions	modificar_reunions	eliminar_reunions	llistar_reunions	afegir_foto
modificar_foto	eliminar_foto	llistar_foto		

Segons el disseny efectuat i el codi generat, jeràrquica i funcionalment, el Sistema de Gestió de la Informació està dividit en tres seccions: (1) gestió del director, (2) gestió del tutor i (3) gestor fotogràfic.

- El **gestor del director** permet:
 - Gestionar professionals. Afegir, eliminar, modificar i llistar. Es pot modificar tot menys la clau d'accés (només el propi usuari pot modificar la clau: és personal i per responsabilitat pròpia de les accions).
 - Gestionar alumnes. Afegir, eliminar, modificar i llistar.
 - Gestionar aules i classes. Es defineix **aula** com aquell espai físic que roman any rere any i que acull les **classes** que cada any són diferents. Per exemple, "Eclipsi" és el nom d'una aula (espai físic que s'hi accedeix per unes portes, etc.) i que cada curs acadèmic acull una classe diferent. Per exemple, el curs 2012/13 va acollir una classe formada per un tutor, uns alumnes, etc., que va existir aquell curs acadèmic i que mai tornarà a existir. La classe té un identificador únic (no es tornarà a repetir més) anomenat: Eclipsi, curs 2012/13. El gestor permet crear, eliminar, modificar i llistar **classes**. Un cop la classe està creada, es pot gestionar els tutors i alumnes que en formen part. Aquests paràmetre també es poden modificar en qualsevol moment. Les **aules** es creen a partir del nom de l'aula i el curs. Els noms d'aules es poden crear i eliminar.
 - Exemple: es pot crear l'aula "Eclipsi, 2015-16" a partir d'"Eclipsi" i el curs "2015-16".
 - Gestionar normes. Afegir, eliminar, modificar i llistar.
- El **gestor del tutor** permet:
 - Gestionar faltes d'assistència. Afegir, eliminar i llistar.
 - Gestionar incidències. Afegir, eliminar i llistar.
 - Gestionar informes. Afegir, eliminar, modificar i llistar.
 - Gestionar reunions. Afegir, eliminar, modificar i llistar.
- El **gestor fotogràfic**:
 - Gestionar fotografies. Afegir, modificar, eliminar.
 - Visualitzar el catàleg de fotografies i possibilitat de filtratge per aula, data i/o etiqueta.

En funció del rol de l'usuari es pot accedir a una o més seccions. El rol de director permet accedir a totes les seccions, el rol de tutor permet accedir a la secció "gestió del tutor" i "gestor fotogràfic", i el rol d'usuari normal només permet accedir al darrer. Un usuari tutor només pot gestionar faltes d'assistència, incidències, informes i reunions dels alumnes que estiguin donats d'alta en classes seves. Un usuari director podrà llistar totes les assistències, incidències, informes i reunions, però no les podrà gestionar. Només podrà fer-ho l'usuari tutor qui les va crear.

Adicionalment a aquestes funcionalitats, el Gestor també permet gestionar **opcions de l'usuari**. Aquestes són canviar la fotografia, canviar la clau d'accés i tancar la sessió. Cada usuari és responsable de les accions que dugui a terme, les quals són registrades. Per tant, és important mantenir la confidencialitat de la clau d'accés (personal i intransferible) i tancar la sessió un cop finalitzada la feina.

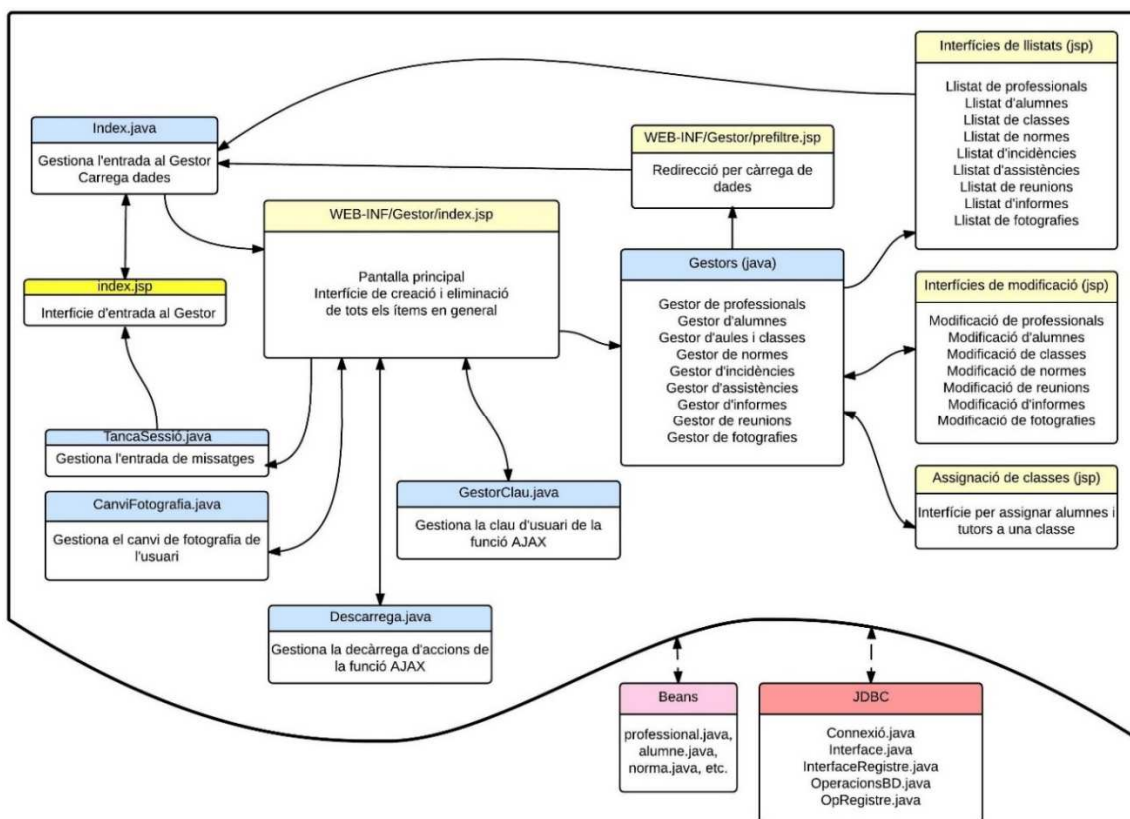
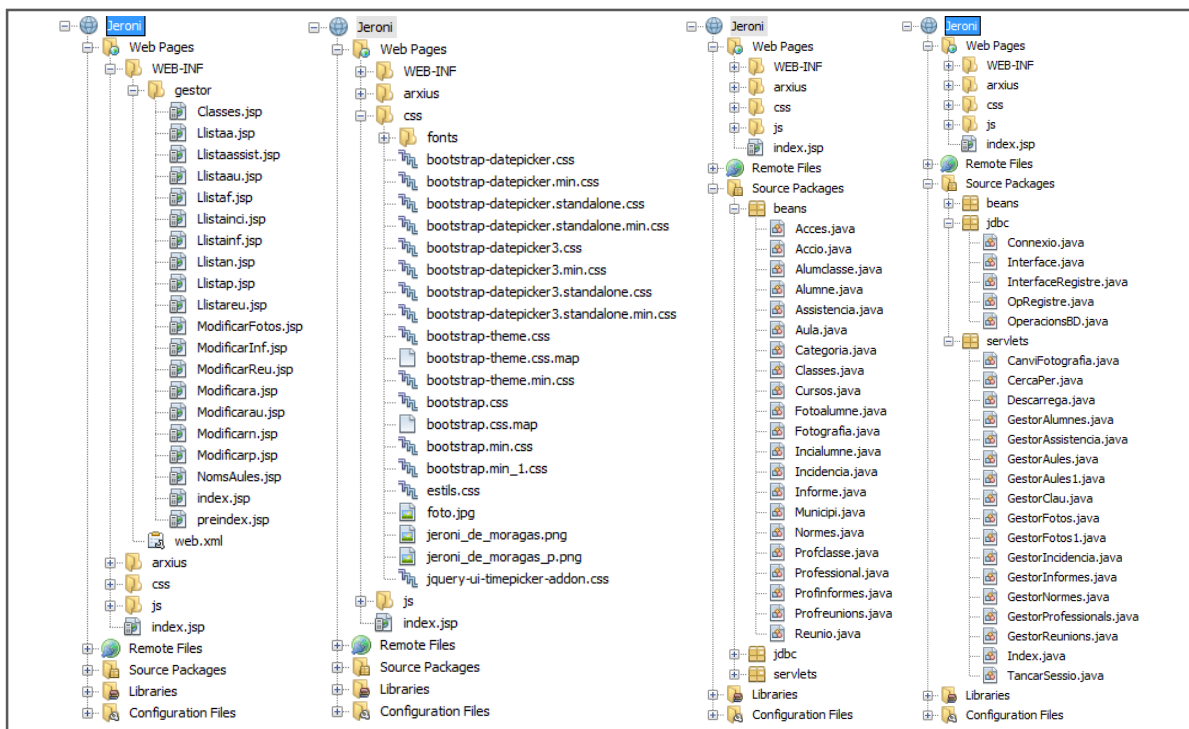


Figura 26. Estructura del codi i projecte del Sistema de Gestió de la Informació

El gestor té incorporat un primer sistema de comunicació amb l'usuari. Aquest consisteix en informar-lo després d'una acció de creació, modificació o eliminació. S'informa si s'ha efectuat amb èxit o no. En una segona fase, es podrà millorar aquest sistema incloent una descripció del motiu pel qual no s'ha realitzat amb èxit l'operació, si és que ha estat així.

En relació al disseny gràfic, s'ha considerat que és una aplicació de treball i no un programa de projecció exterior. Per tant, és un disseny molt sobri i clar també d'acord amb els requeriments de l'usuari. El disseny està basat en:

- Dos únics elements gràfics: el logotip d'AJEM i la fotografia de l'usuari
- Tipografia amable i de color gris fosc evitant el negre.
- Botons estilitzats segons la mateixa tonalitat de gris que combina amb el text i aporta color (defugint de l'aspecte tradicional dels botons de formularis web).
- L'efecte de l'aparició d'opcions quan es clica als títols de les opcions. Proporcionar un toc tecnològic modern.

Adicionalment al desenvolupament de l'aplicació, també s'ha desenvolupat una segona petita aplicació d'escriptori (també programada en **Java** i amb l'**IDE Netbeans** i en aquest cas usant la llibreria del *swing* que l'**IDE** porta incorporada per defecte) per gestionar les còpies de seguretat de la informació continguda a l'aplicació: tant les dades de la SGBD com els arxius desats en carpetes (figura 27).

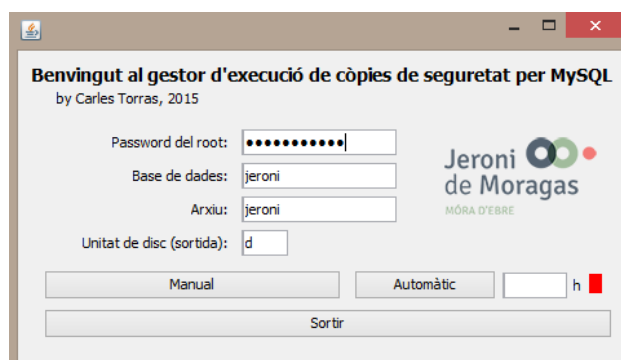


Figura 27. Interfície de l'aplicació de còpies de seguretat del Sistema de Gestió de la Informació

L'aplicació (que no requereix instal·lació) es desarà i s'executarà en l'ordinador servidor del CEE, on hi haurà instal·lada la BD i el servidor web *Glassfish* amb l'aplicació del Gestor i on hi ha instal·lat actualment el programa de còpies de seguretat.

L'aplicació genera un arxiu tipus *dump sql* de l'estat actual de la base de dades del Gestor (taules + dades) i un arxiu ZIP amb tots els arxius relatius al Gestor i que són referenciats en la BD. L'aplicació desa ambdós arxius a una carpeta del disc dur del mateix ordinador servidor sobre la qual l'actual programa de còpies de seguretat del CEE (*Cobian Backup*) en realitza còpia. L'aplicació permet realitzar una generació d'arxius en un moment puntual o programar que se'n realitza una a cada espai de temps especificat (mitjançant *javax.swing.Timer*). Donat el relatiu baix volum d'operacions i dades que s'espera i els recursos disponibles, es considera una solució prou adequada.

L'aplicació conté formularis per poder canviar la configuració. Tanmateix, per defecte, ja porta incorporada l'específica per aquest cas (establerta en la fase d'implantació). Addicionalment, quan s'executa el programa, ja s'inicia automàticament la programació automàtica d'acord amb el protocol de còpies de seguretat d'AJEM. Així, davant d'un possible reinici del sistema o del programa, es garanteix el funcionament del gestor.

Execució de proves

El servei del Sistema de Gestió de la Informació es subministra mitjançant un servidor intern del CEE. Tanmateix, per la fase de proves, es va instal·lar en un servidor extern (el mateix del Bloc) el qual oferia disponibilitat des de qualsevol màquina connectada a Internet. Per aquesta finalitat i per no incomplir la Llei de Protecció de Dades es van usar dades fictícies. Així, un cop l'aplicació va estar desenvolupada i provada localment per l'autor, es va activar una versió de proves en el servidor provisional de proves, alimentat amb dades fictícies (<http://jeroni.necessito.cat>).

L'execució de proves es va iniciar el 06/05/2015 i va finalitzar el 27/05/2015. Durant aquest temps, els professionals del CEE i d'altres actors van anar provant l'eina, detectant errors i millores a fer en quan a funcionalitat i usabilitat, les quals van ser totes considerades i aplicades. Un resum de les mateixes són (canvis o afegits respecte al disseny inicial):

- Possibilitat d'afegir diversos tutors en una classe.
- Afegir un atribut extra pels professional per indicar si és un usuari actiu o no.
- Que en la gestió d'incidències, el tutor tingui disponible tota la llista d'alumnes i no només els que ell/a tutoritza.
- En el catàleg de fotografies, afegir etiquetes d'alumnes a les fotografies a partir de desplegable selectors. Afegir també el filtre d'alumne en el catàleg.
- Diverses millores de presentació del Bloc (*layout*).
- Correcció d'errors.

5.4.- Documentació

En l'execució del present projecte final de màster, s'han generat diversos tipus de documentació. Aquests es poden classificar segons el client, distingint-ne dos: el CEE Jeroni de Moragas i la UOC.

En relació al CEE Jeroni de Moragas, s'han preparat quatre documents els quals es comenten a continuació, a banda de les actes de les diverses reunions celebrades (5 fins al moment):

- **I001.** Aquest primer document correspon a la Política d'IT del CEE (lliurable planificat). El document estableix els programes que s'usen en els diversos ordinadors del centre i la seva justificació. El document també ha de servir de guia a usar i normativa a complir en la posada a punt dels nous ordinadors que el CEE adquireixi. És un document viu que cal que vagi evolucionant en el temps, a partir de les noves necessitats i nou programari de base disponible.

- **I002.** Aquest document correspon a la Guia d'ús del Bloc Educatiu. En ella es descriu la funcionalitat de l'eina i com usar-la, pas a pas i detalladament per totes les opcions. Aquesta guia està a disposició de tots els professionals del CEE que poden usar el Bloc (en format electrònic i paper) i també és el material que s'usa en la formació.
- **I003.** Aquest document correspon a la Guia d'ús del Sistema de Gestió de la Informació. En ella es descriu la funcionalitat de l'eina i com usar-la, pas a pas i detalladament per totes les opcions. Aquesta guia està a disposició de tots els professionals del CEE que poden usar el Gestor (en format electrònic i paper) i també és el material que s'usa en la formació.
- **I004.** Aquest document correspon a la Guia d'ús del nou Programari del CEE. En ella s'identifiquen tots els programes que s'usen, perquè serveixen i pretén ser una breu guia molt bàsica de les principals funcionalitats dels ordinadors a partir del programari instal·lat. La guia s'acompanya de guies complertes existents de cadascun dels diversos programes: guia del *Linux Mint*, dues guies del *LibreOffice* (una en anglès de la darrera versió i una en castellà de la darrera versió de la qual es va elaborar la traducció), guia del *Gimp* i dues guies de l'*Audacity*. El fet que elaborar una guia per aquests programes és quelcom complex i llarg i el fet que ja n'existeixen d'oficials i de qualitat, han justificat aquesta combinació. Aquesta guia està a disposició de tots els professionals del CEE que poden usar el Gestor (en format electrònic i paper) i també és el material que s'usa en la formació.

A l'annex 5 d'aquest informe s'adjunten tots aquests documents.

En relació a les actes, l'objectiu fonamental de cadascuna és documentar els diversos aspectes tractats, així com els acords assolits.

En relació a la UOC, es requeria el lliurament dels següents documents, els quals ja han estat lliurats a excepció dels darrers de cada assignatura que estan en fase de preparació:

- Assignatura Treball Final de Màster. PAC 1: Estudi de viabilitat i anàlisi del sistema. Termini de lliurament: 11/03/2015.
- Assignatura Treball Final de Màster. PAC 2: Disseny del sistema. Termini de lliurament: 15/04/2015.
- Assignatura Treball Final de Màster. PAC 3: Desenvolupament i implantació. Termini de lliurament: 27/05/2015.
- Assignatura Treball Final de Màster. PAC 4: Memòria i defensa. Termini de lliurament: 17/06/2015. Aquest document inclou un compendi dels anteriors documents de l'assignatura, una presentació amb diapositives i un vídeo.
- Assignatura Pràctiques externes. PAC 1: Pla de treball. Termini de lliurament: 11/03/2015.
- Assignatura Pràctiques externes. PAC 2: Informe final + Projecte. Termini de lliurament: 10/06/2015.

6.- Implantació

La figura 28 mostra la planificació de l'etapa d'implantació (ampliada a l'annex 1). Aquesta etapa afecta a gairebé totes les tasques. La planificació (tant cronograma com dedicació) s'ha efectuat a partir de la general, efectuada en la planificació del projecte. A l'entrar en detall, degut a les activitats ja realitzades fins ara i l'estat avançat del projecte, existeixen petites adaptacions i canvis.

En relació a les tasques de creació de serveis (T2.1 – Bloc Educatiu i T2.2 – Gestor Informàtic) la planificació de la implantació és molt similar en les dues i consisteix en iteracions de: (1) presentació/formació dels serveis, (2) període de proves per part dels professionals i (3) correccions d'errades i aplicació de millores. D'acord amb les anteriors etapes, primer es realitza el procés pel Bloc i després pel Gestor.

Tanmateix, una diferència rellevant és que pel cas del Gestor, es realitza molt al principi una presentació i validació només de la Base de Dades. És fa així perquè en aquest cas el disseny de la base de dades és crític sobre el desenvolupament del codi de l'aplicació. És a dir, un canvi posterior en la base de dades, pot suposar un sobre-esforç significatiu en la modificació del codi. Pel cas de Bloc, no és així degut a què és molt més simple.

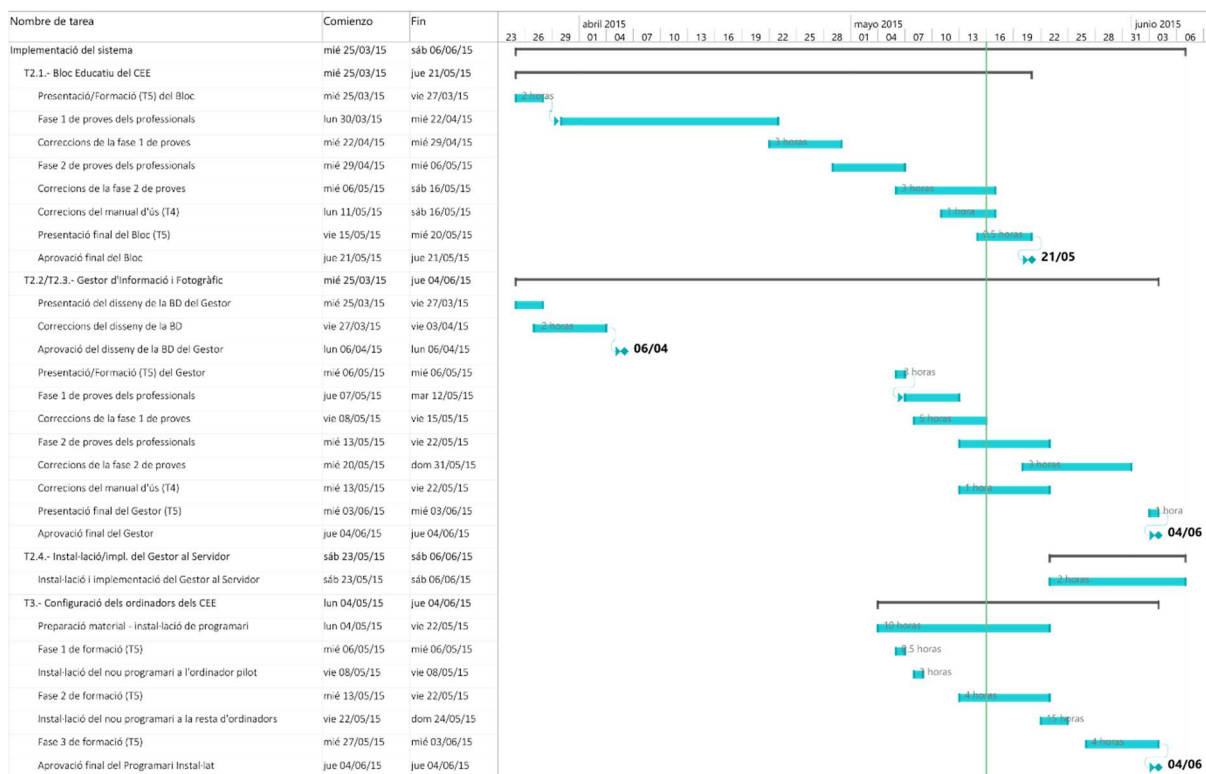


Figura 28. Cronograma de la fase d'implementació

En quan a la T3, relativa a la implantació del nou programari de base, la planificació té en compte que el desplegament del nou programari s'ha d'iniciar només quan ja es disposi de guies d'ús i s'hagi realitzat formació als professionals del CEE. També cal que s'hagi preparat convenientment el material per instal·lar-lo i tenir també el necessari per afrontar contingències. Primer es realitza la instal·lació en un ordinador pilot, la qual cosa facilita comprovar compatibilitat de maquinari (del propi ordinador i dels altres) i permetre als professionals començar a provar el nou sistema.

Les actuacions al CEE s'han planificat de manera que afectin el menys possible a la rutina del centre. Per aquest motiu, s'han agrupat actuacions en mateixos dies i fora del horari lectiu (incloent dissabtes). Per exemple, l'adaptació de l'ordinador servidor que allotjarà el gestor es realitza el mateix dia que es realitza la implementació del nou programari de base a la majoria d'ordinadors del CEE.

Finalment, comentar que en el diagrama de Gantt no s'han inclòs els recursos humans ja que tot ho realitza una única persona (l'autor). Totes les hores de dedicació indicades en les barres són a càrrec del mateix. L'única excepció són les tasques de proves, que són executades pels professionals del centre (s'entén que són recursos i tasques externes al projecte i no s'hi assignen hores).

6.1.- Formació

D'acord amb la planificació, s'han realitzat diverses etapes de formació al CEE en relació als diversos serveis a implantar. En tots els casos, la metodologia de formació ha estat la següent:

- Formació presencial en reunions periòdiques les quals han tingut lloc diversos dimecres, de 17-18:30h al CEE. En cadascuna de les reunions/formacions s'ha elaborat acta de la reunió (adjuntes a l'annex 2), indicant-hi també l'assistència (que ha estat pràcticament total en tots els dies). El CEE està format per 8 professionals de promig.
- Per cadascuna de les formacions, l'autor havia preparat material didàctic (documentació especificada en l'apartat 1.4 d'aquest informe), el qual s'havia repartit als professionals del CEE, prèviament a la formació, en suport paper (físic) i posteriorment també en PDF (suport electrònic).
- Així, en cada formació, es treballava amb:
 - Explicacions de l'autor i discussió amb els professionals del CEE.
 - Material didàctic comentat.
 - Suport multimèdia. Les reunions s'han efectuat en una aula que disposava d'una pantalla gegant en la qual es projectava el recurs informàtic escaient (les aplicacions desenvolupades, el sistema operatiu *Linux Mint* i les aplicacions contingudes, etc.).

El temes objectes de formació han estat els següents:

- **Abast del projecte i justificació.** En primer lloc, calia fer comprendre a tots els professionals la necessitat d'executar aquest projecte, quins problemes existeixen en la situació inicial (manca de base de dades electrònica del CEE, impossibilitat de gestió de dades, dificultat d'accés a la informació, incompliments legals, manca de compatibilitat, desorganització general, etc.), quins avantatges se'n derivarien i també quins inconvenients hi hauria (per exemple, necessitat

d'adaptar-se el canvi). Amb això es va aconseguir també una bona implicació de tots els professionals.

- **Conceptes bàsics.** El desconeixement d'aspectes informàtics entre els professionals del CEE era important i a diversos nivells. Entre d'altres:
 - Diferència entre dades i arxius i les implicacions d'això. Gestió de dades.
 - Què és un sistema operatiu, quins tipus hi ha, història, etc.
 - Diferència i implicacions de llicències privatives, llicències gratuïtes o llicències de codi obert.
 - Definició d'interfície. Diferències i similituds entre aplicacions d'escriptori i aplicacions web.
- **Bloc Educatiu.** Funcionament i ús d'aquesta aplicació.
- **Sistema de Gestió de la Informació.** Funcionament i ús d'aquesta aplicació.
- **Nou programari de base.** En aquest apartat s'inclouen només aspectes bàsics de diversos programes. L'objectiu era oferir una introducció que permetés als usuaris afrontar el canvi amb una certa garantia. Es requeririen un gran nombre d'hores per aprofundir en cadascun dels programes. Específicament s'ha tractat el següent:
 - **Linux Mint** [11]. Conceptes bàsics dels sistemes GNU/Linux (per exemple, concepte de permisos). Ús bàsic del SO: com entrar al sistema, com parar-lo, com navegar-hi, com executar aplicacions, com accedir als arxius, etc.
 - **LibreOffice** [7]. Conceptes bàsics d'ús. I en particular i a partir de la pròpia demanda dels professionals: com generar formularis, com treballar amb PDF, compatibilitat entre diversos processadors de textos.
 - **Gimp** [18], **Audacity** [19], **Brasero**. Conceptes bàsics d'ús.

6.2.- Implantació del sistema i proves

A continuació, es detalla el procés d'implantació del tres subsistemes, els quals tenen característiques prou diferenciades.

6.2.1.- Bloc Educatiu

Les proves amb el Bloc Educatiu es van realitzar a dos nivells:

- Proves a nivell del desenvolupador. Proves realitzades en el mateix entorn de desenvolupament (*IDE Netbeans 8.0*), la qual integra base de dades i servidor web.
- Proves a nivell d'usuaris. Aquestes proves ja es van realitzar en el seu entorn operatiu definitiu, però amb accés restringit (<http://blocjeroni.necessito.cat>)

Així, donada la manera com es va realitzar el darrer nivell de proves, el pas a nivell de producció ha estat, en aquest cas, obvi. La seva operativitat ja es va garantir. La única acció que ha calgut realitzar és eliminar el control d'accés (qualsevol usuari que accedeixi al Bloc mitjançant l'anterior URL, visualitzarà les entrades).

Les dades inicials requerides en aquest cas només han estat els noms de les aules i els usuaris del Bloc (aquells amb capacitat d'introduir nous missatges i administrat el Bloc). Les mateixes dades que es feren

servir per les proves en relació a aquests conceptes són les existents en l'entorn de producció. Per tant, no s'ha requerit tampoc canvis en la BD.

L'esquema de base de dades per aquest servei té el nom i l'estructura de taules dissenyat en les fases anteriors de disseny i proves. L'esquema s'ha importat a la BD de l'entorn operatiu definitiu mitjançant un arxiu SQL. En la SGBD s'ha creat el nom d'usuari i contrasenya propis per l'esquema.

6.2.2.- Sistema de Gestió de la Informació

Les proves amb el Gestor es van realitzar a dos nivells:

- Proves a nivell del desenvolupador. Proves realitzades en el mateix entorn de desenvolupament (*IDE Netbeans 8.0*), la qual integra base de dades i servidor web.
- Proves a nivell d'usuaris. Aquestes proves es van realitzar instal·lant el servei en un servidor extern proveït de *MySQL*, *Java* i *Glassfish*. L'accés es realitzava a través de la URL <http://jeroni.necessito.cat>. Totes les dades usades en aquesta fase van ser dades fictícies.

Per tant, en aquest cas, sí que s'ha requerit una etapa d'implantació a l'entorn operatiu definitiu. Aquest corresponia al servidor d'AJEM. Es van preveure i s'han realitzat les següent operacions:

- Adequació del servidor. La màquina no disposava d'entorn *Java* ni *Glassfish*. Ambdós serveis s'han hagut d'instal·lar.
- Implantació del programa de gestió de còpies de seguretat. S'ha requerit la gravació del programa, la seva configuració (establiment de les rutes d'arxius fons i destí) i proves en entorn d'operació definitiu.
Una vegada establerta la seva configuració, les dades queden desades en el programa de manera que davant d'un reinici del mateix, no cal tornar a configurar-lo. Tanmateix, es poden canviar.
- Càrrega de dades inicials. Per aquest servei, el volum de dades inicials és gran i s'ha previst una metodologia especial. És un procés que s'ha iniciat després de la implantació del servei i que s'allargarà fins a setembre de 2015, quan entrarà en funcionament oficial el Gestor. La preparació de les dades va a càrrec dels professionals del CEE i la seva introducció a la BD a càrrec de l'autor.

Així, la metodologia per carregar les dades inicials no és la prevista per la càrrega de dades en operació normal del gestor, la qual es realitza a través dels formularis del mateix. En aquest, les dades es bolquen directament dins la BD usant instruccions SQL executades dins de *MySQL Workbench* instal·lada en l'entorn operatiu definitiu. L'autor ha facilitat una plantilla als professionals del centre (arxiu de text pla, el qual poden treballar en qualsevol ordinador, en qualsevol moment i en local) sobre la qual els professionals van omplint les dades. Posteriorment, l'autor les introdueix a la BD.

L'esquema de base de dades per aquest servei té el nom i l'estructura de taules dissenyat en les fases anteriors de disseny i proves. L'esquema s'ha importat a la BD de l'entorn operatiu definitiu mitjançant un arxiu SQL. En la SGBD s'ha creat el nom d'usuari i contrasenya propis per l'esquema.

Un cop implantat el sistema, s’han realitzat diverses proves *in-situ*, experimentant també un reinici de la màquina servidor (tot i que és poc probable degut a la presència del SAI). Les diverses proves han resultat satisfactòries.

6.2.3.- Nou programari de base

El procés d’implantació del nou programari ha seguit una sèrie de passos, segons estava planificat. A diferència dels nous serveis (Bloc i Gestor corresponents a uns serveis webs que s’afegien al sistema del CEE), aquesta acció ha suposat un canvi obligat en la rutina de tots els professionals. Per tant, ha requerit un procés curós per facilitar en canvi i evitar situacions traumàtiques per part de l’usuari.

La figura 29 mostra el procés d’implantació d’aquest subsistema.

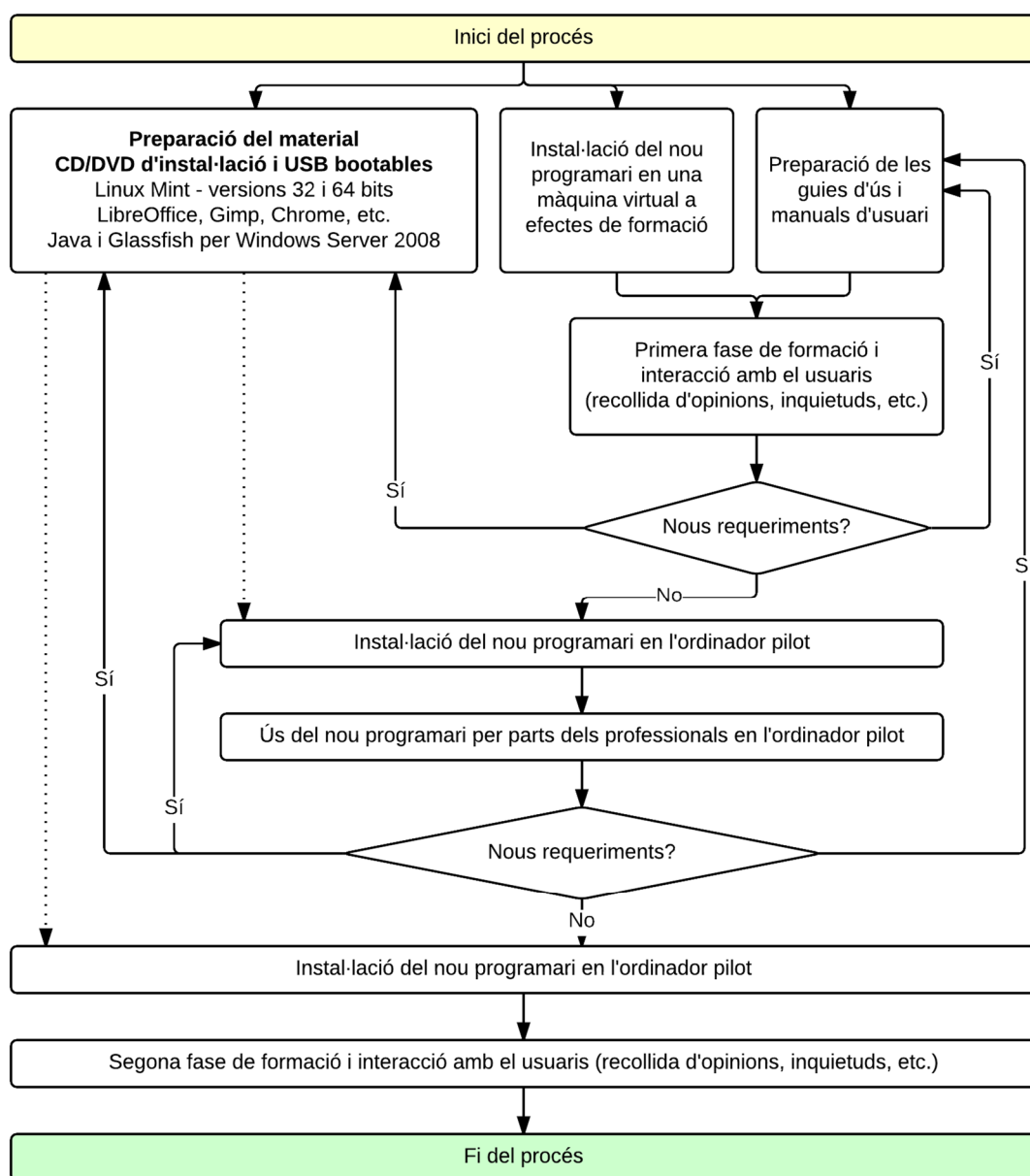


Figura 29. Procés d’implantació del nou programari de base

Així, abans d'actuar sobre el primer ordinador del CEE (el pilot), s'han enllestit les guies d'ús, s'han distribuït als professionals del CEE i s'ha realitzat una primera formació.

Com mostra la figura, la instal·lació del nou programari de base s'ha realitzat en tres plataformes diferents, amb diversos objectius, de menys compromís a més i avançant a mesura que existia l'aprovació dels professionals del CEE. En cadascuna d'aquestes etapes s'han atès nous requeriments i imprevistos, prioritant els objectius del CEE i facilitant el procés als professionals.

L'etapa més crítica ha correspost a la següent després d'haver instal·lat el nou programari de base en l'ordinador pilot del CEE. Han confluït dos factors: el fet que els professionals han començat a treballar realment amb el nou sistema i que durant aquest període s'ha iniciat (avançadament) el període de final de curs al CEE. Això suposa una sobre-càrrega de treball als professionals i una situació més estressant.

Aquests fets van provocar la petició d'allargar la fase d'implantació del nou programari de base al CEE. És a dir, en alguns ordinadors claus, no realitzar la implantació fins passada l'etapa de final de curs. Es van valorar dos aspectes: les necessitats del CEE i els seus professionals i les necessitats del propi projecte en quan a TFM de la UOC. Entenent que el TFM representa l'execució d'un projecte real amb successos que impliquen re-planificar fites i destinat a cobrir les necessitats del client (el CEE en aquest cas), es va decidir prioritzar les necessitats del CEE. En conseqüència, aprovar la re-planificació. En tot cas, també es va valorar que el nou programari de base quedaria instal·lat en 5 de les 8 màquines del CEE. Per tant, en una majoria.

Així, s'ha implementat el nou programari de base en 5 màquines (Logopèdia, Eclipsi, Lluna Nova A, ordinador alumnes Lluna Plena i ordinador alumnes Quart Creixent). I en les 3 restants (Lluna Nova B, ordinador professional Lluna Plena i ordinador professional Quart Creixent), el procés es realitzarà a partir de finals de juny, un cop finalitzada la fase de final de curs del CEE.

Durant el procés d'implantació tant sols ha sorgit un inconvenient. La màquina de l'aula Logopèdia no ha estat capaç de suportar amb suficient solvència el sistema operatiu *Linux Mint*, en cap de les versions d'escriptori disponibles (incloent les més lleugeres). Aquest risc ja estava contemplat en la planificació (i degut a la vellesa de l'ordinador) i s'ha aplicat el pla de contingència previst: deixar-hi implantat el SO que hi havia (Windows XP) però actualitzant tot el programari al nou definit com en la resta de màquines del CEE (LibreOffice, Gimp, Audacity, etc.).

Un cop realitzada l'anterior implantació, s'ha procedit a realitzar la segona etapa de formació.

6.3.- Nivell de servei

La planificació del projecte i el disseny dels subsistemes es va realitzar considerant com a bases que els nous serveis i programari fossin el màxim autosuficients i autònoms possibles i que requerissin els mínims nous serveis possibles, una vegada finalitzada la implantació. La justificació principal és que el CEE no disposarà de recursos extres per destinar-hi. Tanmateix i en relació a cada subsistema, es realitzen les següents consideracions.

6.3.1.- Bloc Educatiu

El bloc educatiu està allotjat en un servei extern.

Aquest servei extern ofereix al CEE l'allotjament, ús de domini i gestió de còpies de seguretat dels continguts del Bloc. Així, també inclou el manteniment del servei prestat i suport sense compromís en quan a temps de resposta, mitjà i d'altres.

A canvi, el prestador no rep cap contraprestació per part del CEE (servei de cortesia). Però alhora, el prestador s'eximeix de qualsevol responsabilitat.

No s'ha realitzat cap contracte o conveni formal, ni s'ha acordat cap temps de vigència de l'acord (per tant, tampoc cap data de caducitat de prestació del servei). Si per algun motiu es deixa de donar el servei o el CEE vol deixar de rebre'l, el prestador lliurarà al CEE tot el material contingut en el bloc i l'eliminarà de la seva base de dades.

Aquests termes estant acceptats per totes les parts. En cas de finalització de la prestació de serveis, el CEE valorarà com procedir. Per continuar gaudint del servei del Bloc, es podrien estudiar dues opcions: allotjar el servei en el propi centre (caldrà estudiar la conveniència d'usar l'actual màquina servidor o una altre) o per exemple, allotjar el Bloc en un altre servei extern, com el que actualment allotja la web d'AJEM (servei de pagament).

En qualsevol cas, les necessitats del servei del Bloc són escasses. Es requereix un servidor web amb funcionalitat per Java i cert espai de disc dur, que dependrà de l'ús del Bloc. Donat el número de persones que tenen possibilitat de realitzar entrades, la intensitat de la seva dedicació en realitzar-les i la reducció automàtica de la mida de fotografies que s'incorporen, es preveu un nivell baix de requeriment d'emmagatzematge. En termes de seguretat, els requeriments són baixos ja que la informació continguda en el bloc és pública i no conté tipus de dades considerades per la Llei de Protecció de Dades.

6.3.2.- Sistema de Gestió de la Informació

El Sistema de Gestió de la Informació està allotjat en el servidor d'AJEM, al qual s'hi va instal·lar *Java*, el servidor web *Glassfish 4.0* (versió lliure) i el gestor de còpies de seguretat. Aquests són els únics requeriments del servei, a part d'espai d'emmagatzematge per les dades.

Sobre l'emmagatzematge, es preveu una necessitat baixa, malgrat també està en funció del seu ús. La previsió es basa en la mida de centre, format per 6 aules, el mateix ordre de professionals i, en general, no hi ha més de 10 alumnes per aula. Així, el volum de dades que es generarà serà baixa. Tanmateix, també cal considerar els arxius, que malgrat no es desen dins la base de dades (només els enllaços) també formen part de la necessitat d'emmagatzematge. Hi ha dos tipus d'arxius, les fotografies i arxius essencialment de text. Els darrers són de mida molt reduïda. I pels primers, el servei també n'optimitza la mida (balancejant-la amb la qualitat) i no permet afegir més d'una fotografia al mateix temps. Això es va dissenyar expressament per estimular a l'usuari desar només fotografies seleccionades.

L'aplicació de còpies de seguretat realitza automàticament (i amb una freqüència programada) una còpia del contingut de la base de dades i de tots els arxius associats (empaquetats en un contenidor ZIP) a una carpeta. Alhora, d'aquesta carpeta, el programa de còpies de seguretat d'AJEM en realitza còpies segons el procediment establert i vigent en l'Associació.

Per tant, no es requereix cap servei específic més enllà de la disponibilitat d'espai de disc dur (que amb l'espai actual es preveu que el servei està garantit durant molt temps) i una supervisió esporàdica de què els processos descrits es realitzen correctament. Actualment, aquesta supervisió ja existeix en els serveis actuals (la realitza el gerent d'AJEM) i per tant, pràcticament no s'incrementen els serveis requerits.

En quan a seguretat, les dades contingudes en el Gestor són molt sensibles en relació a la Llei de Protecció de Dades. En conseqüència, se'ls hi realitza el mateix tractament i consideració que altres dades del mateix nivell que ja existeixen a AJEM. Per exemple, el Gestor i les dades només són accessibles dins la xarxa interna d'AJEM. Un assessor extern d'AJEM en supervisa el seu tractament d'acord amb la LOPD.

6.3.3.- Nou programari de base

La base de disseny en relació al nou programari de base fou que no es creessin més serveis necessaris dels que hi ha actualment. Els serveis actuals són externs i consisteixen en enviar un ordinador a reparar quan es danya el sistema. Això mateix es requerirà en aquest cas. Tanmateix, donades les característiques dels sistemes GNU/Linux en comparació amb els de Microsoft® Windows® [3], es preveu que les incidències relacionades amb aquest aspecte disminueixin.

El projecte ha inclòs un estudi del programari necessari al CEE i en conseqüència, no s'haurien de requerir més necessitats en els ordinadors. Per tant, la variació principal que hi haurà serà la pròpia de creació de nous arxius d'usuari. El maquinari actual del CEE disposa de capacitat per assumir aquest increment durant un període llarg de temps. I addicionalment, durant l'execució d'aquest projecte s'ha adquirit un nou disc dur extern de 2 TB que ajudarà a satisfer aquestes necessitats.

Per totes les possibles altres petites intervencions que es requereixin, els propis usuaris n'assumiran l'execució tal i com ha vingut succeint fins ara.

Finalment, un aspecte a destacar i que serveix per tots els subsistemes és que l'autor d'aquest projecte, malgrat finalitzi el present TFM, continuarà donant suport al CEE per garantir el correcte desplegament final de tots els serveis i nou programari de base. Es preveu que aquest suport tindrà una durada de diversos mesos, en els quals la intensitat del servei anirà disminuint progressivament.

6.4.- Acceptació del sistema

L'acceptació s'ha efectuat per cadascun dels subsistemes, que ha conduït a l'acceptació final del projecte.

Pel cas del Bloc Educatiu, l'acceptació final s'ha produït en una reunió amb els professionals del CEE el dia 20 de maig de 2015 (veure acta annex 2). S'ha produït un cop finalitzat el procés de prova, esmenats els darrers errors i implementades les darreres millores. També, un cop lliurada i acceptada tota la documentació del subsistema i realitzada la formació.

Pel cas del Sistema de Gestió de la Informació, la pre-acceptació s'ha produït en una reunió amb els professionals del CEE el dia 20 de maig de 2015 (veure acta annex 2). Com en el cas anterior, s'ha produït un cop finalitzades les proves, lliurada i acceptada tota la documentació del subsistema i realitzada la formació. Tanmateix, cal considerar que la posada en marxa del servei serà progressiva fins al setembre de 2015, coincidint amb l'inici del nou curs i havent-hi introduït tota la informació inicial que està en vies d'obtenció per part dels professionals del CEE (serà llavors quan es realitzarà l'acceptació definitiva). Durant aquest temps, l'autor continuarà donant suport al CEE.

Pel cas del Nou Programari de base, l'acceptació s'ha produït en reunió celebrada el 27 de maig de 2015. S'ha lliurat tota la documentació, s'ha realitzat la informació i el nou sistema ha quedat instal·lat en la majoria de màquines del CEE. Com està explicat a la secció d'implantació, manca per instal·lar el nou sistema a un número reduït de màquines, la qual cosa s'enllestirà entre juny i principis de setembre de 2015.

7.- Manteniment

El manteniment requerit es pot analitzar des del punt de vista de cadascun dels productes desenvolupats: Bloc Educatiu, Sistema de Gestió de la Informació i nou Programari de base. Però també pel tipus: *update* o *upgrade*: el primer fa referència al manteniment de les versions actuals de programari (en el qual també s'inclou en aquest cas les dades) i el segon a la instal·lació de noves versions.

Pel cas del **Nou Programari de base**, algunes necessitats d'*update* simples poden ser cobertes pels propis usuaris mitjançant l'ús del compte d'administrador i l'ús dels assistents d'instal·lació de nou programari i controladors del propi *Linux Mint* (els usuaris tenen la informació del compte d'administrador i se'ls ha format per això). Exemples d'aquests casos poden ser la configuració d'una nova xarxa WIFI, la instal·lació d'una nova impressora o d'un programa Linux. Per aspectes més complexos d'*update* com una fallada del sistema, la solució seria la mateixa que pels casos d'*upgrade*, i que alhora és la mateixa que s'utilitza en l'actualitat: l'assistència d'un servei extern. Actualment, quan per exemple un ordinador s'avaria, es porta a reparar a un servei extern de la mateixa població. Així, caldria procedir de la mateixa manera. En tot cas, el nou sistema té com a base de disseny minimitzar el número de fallades respecte al sistema anterior, sobretot degudes a manipulació incorrecte d'arxius de sistema per via interna (per exemple, accions involuntàries dels usuaris) o externa (per exemple, virus). En relació a l'*upgrade*, la política del CEE ja contempla que amb el pas del temps i la corresponent aparició de nou programari o noves versions i noves necessitats, calguin intervencions d'aquest tipus realitzades de manera procedimental i seguint la pròpia estratègia del CEE.

En relació a l'*upgrade pel Bloc i pel Gestor*, aquest suposaria una re-programació del codi. Això requeriria la intervenció d'un tècnic, tant per realitzar-ho com per implantar-ho posteriorment. És possible que en el futur hi hagin nous requeriments que ho facin necessari, i en aquest cas, la solució haurà de ser l'exposada.

En relació a l'*update del Bloc Educatiu*, el servei està allotjat en un servidor extern, el qual es mantingut per qui gestiona el servei (en el capítol d'implantació es comenten els detalls). En conseqüència, aquest servei no requereix manteniment per part del CEE. En cas que aquest s'integrés en el servidor d'AJEM, el manteniment essencial consistiria en realitzar còpies de seguretat de la base de dades i de la carpeta on estant continguts els arxius (fotografies). La base de dades conté les rutes relatives d'aquests arxius respecte l'arrel del Bloc/projecte. Aquest procés es podria integrar dins del programa gestor de còpies de seguretat desenvolupat pel Sistema de Gestió de la Informació. En qualsevol cas, caldria la intervenció d'un tècnic.

Finalment, en relació a l'*update del Sistema de Gestió de la Informació*, el manteniment de les dades s'ha previst en aquest mateix projecte mitjançant el desenvolupament i implantació d'un gestor de còpies de seguretat. Alhora, aquest està lligat amb l'actual sistema de còpies d'AJEM per tal de facilitar el procés.

El programa realitza diàriament una còpia de les dades: (1) es crea un *backup* de la BD en un arxiu SQL, (2) es crea un contenidor ZIP on s'hi desen tots els arxius associats ubicats en la carpeta del servidor

web. També s'hi afegeix l'arxiu SQL anteriorment creat. La còpia diària (ZIP amb el nom especificat) es sobre-escriu cada dia. Però addicionalment, un cop cada setmana, es desa també un arxiu que no es sobre-escriu amb la mateixa informació que els altres.

L'arxiu diari s'anomena per defecte jeronisql.zip, però es pot canviar mitjançant la interfície del programa. L'arxiu setmanal que no es sobreescriu s'anomena jeronisql ddmmaa hhmms.zip (on *ddmmaa* i *hhmms* fa referència al dia i hora en concret). Aquests arxius ZIP queden copiats a una carpeta de la qual el sistema actual d'AJEM (*Cobian Backup*) en fa les còpies de seguretat corresponents.

El programa està instal·lat com un servei del *Windows Server 2008* (de manera que s'executa automàticament quan es reinicia el sistema) i al fer-ho, es posa en marxa també automàticament la periodificació i execució de les còpies. El programa s'anomena CopiaBD.jar i no cal ser instal·lat. Al ser en Java és multi-plataforma. Si s'hagués d'usar en un altre sistema, només caldria desar-lo en una carpeta coneguda i crear el servei al sistema.

Així, el sistema de còpies de seguretat està totalment automatitzat i no cal cap intervenció, més enllà de comprovar periòdicament que tot funciona adequadament (de la mateixa manera que es realitza amb *Cobian Backup* i per la mateixa persona, com està descrit en el capítol d'implantació).

En el cas poc probable que el sistema fallés i es requerís restaurar les dades, caldria una intervenció d'un tècnic (que segurament ja caldria per la pròpia naturalesa de la fallada). El procediment de restauració consisteix en:

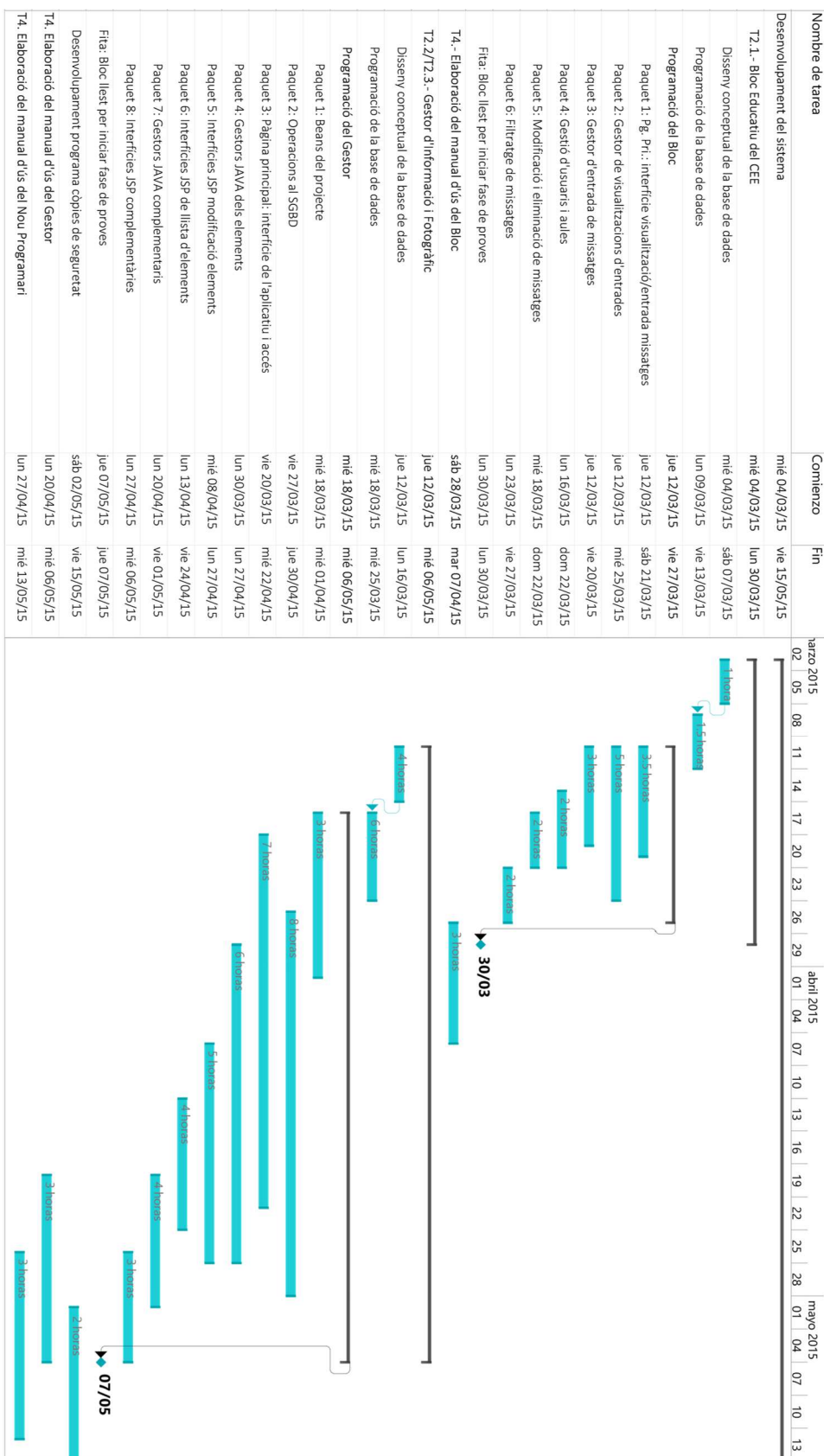
- Restaurar les dades a la BD *MySQL* a partir de l'arxiu SQL contingut en el ZIP de la còpia de seguretat. En funció de si cal restaurar només les dades o també les taules, es pot seleccionar tot l'arxiu SQL o només la part corresponent.
- Copiar la carpeta "arxius" dins del directori arrel de l'aplicació en el servidor web. Podrà ser, per exemple, *C:\GLASSFISH4\GLASSFISH\DOMAINS\DOMAIN1\APPLICATIONS\JERONI*
- En cas de necessitat d'haver de tornar a desplegar l'aplicació al servidor web (*Glassfish v4* en aquest cas), es podria realitzar utilitzant el gestor gràfic del servidor (habitualment accessible mitjançant navegador a la URL local *http://localhost:4848*) i desplegar el contenidor WAR subministrat (arxiu *Jeroni.war*) des de l'apartat d'aplicacions.
- En cas de tractar-se d'un nou sistema, els requeriments de programari pel funcionament del Gestor són *Java v8*, *Glassfish v4* i *MySQL 5.6.19* o superior.

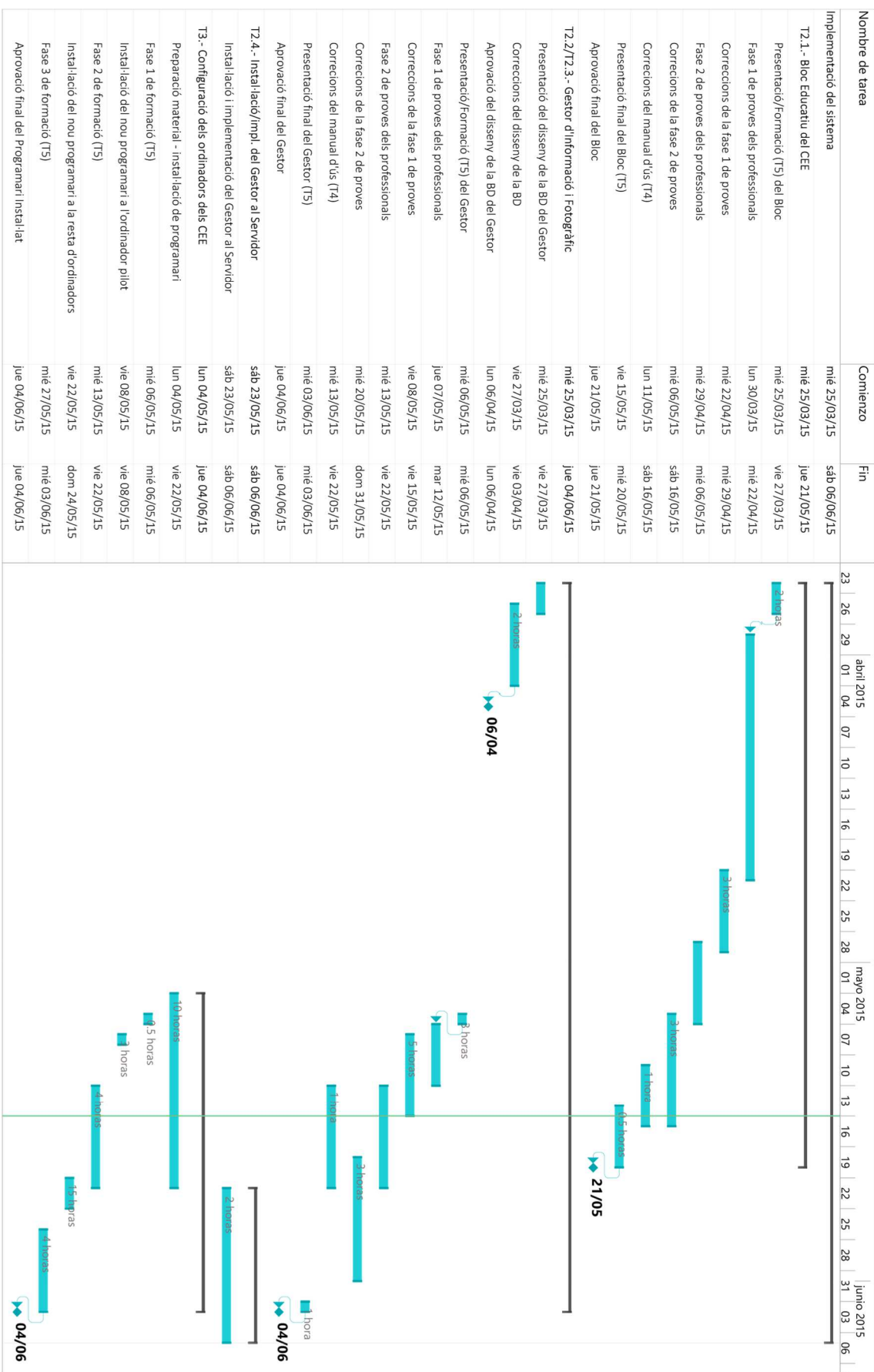
Bibliografia

1. Otero García, A., *Projecte de direcció de sistemes d'informació*, S. Eureka Media, Editor. 2007, Fundació per a la Universitat Oberta de Catalunya.
2. Tate, K. and P. Martin, *The Project Management Memory Jogger*. 2010: GOAL/QPC (Growth Opportunity Alliance of Lawrence).
3. Microsoft. *Pàgina web oficial de Microsoft Windows*. 2015; Available from: <http://windows.microsoft.com/es-es/windows/home>.
4. Digital, N.A. *NUWA Agència Digital*. 2015; Available from: <https://www.nuwa.cat/>.
5. S.L, d. CDMON. 2015; Available from: <https://www.cdmon.com/cas/>.
6. Microsoft. *Pàgina web oficial de Microsoft Office*. 2015; Available from: <https://office.live.com/start/default.aspx>.
7. LibreOffice. *Projecte LibreOffice*. 2015; Available from: <https://ca.libreoffice.org/>.
8. OpenOffice. *Projecte OpenOffice*. 2015; Available from: <http://www.openoffice.org/ca/>.
9. Wikipèdia. *Distribucions de GNU/Linux al llarg dels anys*. 2015 [cited 2015 2015]; Available from: http://ca.wikipedia.org/wiki/Distribuci%C3%B3_de_programari.
10. Canonical. *Distribució Ubuntu*. 2015; Available from: <http://www.ubuntu.com/>.
11. Mint, L. *Distribució Linux Mint*. 2015; Available from: <http://www.linuxmint.com/>.
12. Distrowatch. *Tendències d'ús de distribucions GNU/Linux*. 2015; Available from: <http://distrowatch.com/dwres.php?resource=popularity>.
13. Howtogeek. *Anàlisi de distribucions GNU/Linux*. 2015; Available from: <http://www.howtogeek.com/191207/10-of-the-most-popular-linux-distributions-compared/>.
14. Techradar. *Anàlisi de distribucions GNU/Linux*. 2015; Available from: <http://www.techradar.com/news/software/operating-systems/best-linux-distro-five-we-recommend-1090058>.
15. Everydaylinuxuser. *Anàlisi de distribucions GNU/Linux*. 2015; Available from: <http://www.everydaylinuxuser.com/2014/02/analysis-of-top-10-linux-operating.html>.
16. Sakai. *Projecte Sakai*. 2015; Available from: <https://sakaiproject.org/>.
17. Moodle. *Projecte Moodle*. 2015; Available from: <https://moodle.org/?lang=ca>.
18. GIMP, C.d.d.d. *GIMP. The GNU Image Manipulation Program*. 2015; Available from: <http://www.gimp.org/>.
19. d'Audacity, C.d.d. *Projecte Audacity*. 2015; Available from: <http://audacity.sourceforge.net/>.
20. Oracle. *IDE de desenvolupament Netbeans*. 2015; Available from: <https://netbeans.org/>.
21. d'Eclipse, C.d.d. *IDE de desenvolupament Eclipse*. 2015; Available from: <https://eclipse.org/>.
22. Oracle and C.d.d.d. *MySQL. MySQL Community Edition*. 2015; Available from: <http://www.mysql.com/products/community/>.
23. Group, T.P.G.D. and T.R.o.t.U.o. California. *PostgreSQL*. 2015; Available from: <http://www.postgresql.org/>.
24. UOC, *Normativa per a la presentació del Projecte Final de Màster en Programari Lliure*. 2015.
25. Lucidchart. *Lucidchart*. 2015; Available from: <https://www.lucidchart.com>.
26. Catalunya, G.d. *MuniCat*. 2015; Available from: <http://municat.gencat.cat/index.php?page=descarregues>.
27. Macke, S. *Dia Diagram Editor*. 2015; Available from: <http://sourceforge.net/projects/dia-installer/>.
28. Otto, M. and Jacob. *Bootstrap*. 2015; Available from: <http://getbootstrap.com/>.
29. Foundation, j. *jQuery*. 2015; Available from: <http://jquery.com/>.
30. Edition, C.d.d.G.O.S. and Oracle. *Glassfish Open Source Edition*. 2015; Available from: <https://glassfish.java.net/>.
31. Initiative, O.S., *MIT License*. 2015.

32. Wikipèdia. *Format JSON*. 2015; Available from: <http://es.wikipedia.org/wiki/JSON>.
33. Foundation, T.A.S. *Projecte fileupload*. 2014; Available from: <https://commons.apache.org/proper/commons-fileupload/>.
34. Foundation, T.A.S., *Apache License, version 2.0*. 2004.
35. Sourceforge. *Pàgina web del projecte jexcelapi*. 2015; Available from: <http://jexcelapi.sourceforge.net/>.
36. Free Software Foundation, I., *GNU Lesser General Public License, version 2.1*. 1999.
37. Project, T.F., *The FreeBSD Copyright*. 2015.
38. España, C.C. *Creative Commons España*. 2015; Available from: <http://es.creativecommons.org/blog/cc-es/>.
39. Oracle. *Java*. 2015; Available from: <https://www.java.com/es/>.
40. W3C. *HTML, The Web's Core Language*. 2015; Available from: <http://www.w3.org/html/>.
41. Corporation, O. *JSP Standard Tag Library*. 2013; Available from: <https://jstl.java.net/>.

Annex 1: Cronogrames de les activitats d'integració de sistema







Annex 2. Seguiment de les fases desenvolupament i implantació

Com està detallat en el cos de l'informe (seccions 1.1 i 2), el seguiment de les fases de desenvolupament i implantació s'ha realitzat mitjançant reunions amb els professionals del CEE les quals han generat actes que es presenten a continuació i correspondència entre l'autor i el tutor de la UOC. Aquesta correspondència partia de breus informes preparats per l'autor que s'adjunten a continuació.

Les actes també reflecteixen la formació realitzada als professionals del CEE (secció 2.1).

A2.1.- Actes de reunió

	ACTA DE REUNIÓ – PROJECTE IRENE	AR001_180215_IRENE	
	Reunió: Reunió d'inici de projecte Convocats: Professionals Centre d'Educació Especial Jeroni de Moragas i C. Torras Assistents: Tots els convocats Convoca: C. Torras Secretari: C. Torras	Lloc, data i hora: CEE, Mora d'Ebre, 18/02/15, 17:15-18:30h	

Ordre del dia

- 1.- Presentació del projecte (1h 5min).
- 2.- Prec i preguntes (10 min).

Acta de la reunió

1.- Presentació del projecte

S'inicia la reunió a les 17:15h de la tarda amb les corresponents presentacions i salutacions. La relació d'assistents és la següent:

Anna M ^a Valls, directora del CEE	Raimunda Gris, mestre	Carolina Rofes, mestre
Lina Albiac, mestre	Josep Albiac, mestre	Laura Pérez, mestre
Elisabet Solé, logopeda	Luisa Rodríguez, educadora	África Ruipérez, educadora
Carles Torras		

L'objectiu de la reunió és tenir una primera presa de contacte entre els professionals del Centre d'Educació Especial (CEE) i Carles Torras (CT), explicar el projecte¹ i començar a definir primers passos a seguir.

A continuació, CT explica el context en el qual fa aquesta estada de pràctiques al CEE. Comenta que està estudiant un màster en Programari Lliure (PL) a la Universitat Oberta de Catalunya (UOC) i que una de les assignatures és el Projecte Final de Màster, el qual es pot cursar a partir d'un projecte suggerit per la UOC o cercat pel propi alumne. En aquest cas, donat que CT tenia certa coneixença del CEE i la seva situació, va proposar als responsables del CEE aquest projecte.



A continuació, CT explica resumidament en què consisteix el PL. Explica les principals diferències amb el programari propietari, avantatges i inconvenients, i en quina situació es troba en l'actualitat.

Seguidament, explica quina és la idea del projecte. La situació actual al CEE és que no existeix ni ha existit mai una estratègia de Tecnologies d'Informació (IT). Fins ara, el CEE s'ha alimentat d'ordinadors cedits per diverses entitats i algun puntualment comprat. Així, cada ordinador disposa d'un programari determinat, diferent en cadascun, sense cap política en quan als propis programes, seguretat, etc. Així mateix, el CEE no disposa de cap servei basat en gestió de dades. A partir d'aquí, es proposa dissenyar una estratègia d'IT que contempli:



- Definició de programari oficial del CEE, basat en PL. La idea és que totes les màquines disposin del mateix programari i sigui PL. I que aquest sigui identificat a través d'un procés de pluja d'idees de necessitats d'IT dut a terme per tots els professionals del CEE.
 - El grau d'implementació de nou programari lliure vindrà determinat per l'estudi de necessitats que es realitzarà en la primera fase del projecte.
- Definició de serveis de bases de dades a desenvolupar i implementar.

A partir d'aquí, s'inicia un torn de paraules amb tots els professionals presents, sobretot destinat a clarificar termes. Es tracten diversos punts:

¹ Títol del projecte: "Renovació completa del sistema d'informació intern del Centre d'Educació Especial Jeroni de Moragas", IRENE

	ACTA DE REUNIÓ – PROJECTE IRENE	AR001_180215_IRENE	
	Reunió: Reunió d'inici de projecte Convocats: Professionals Centre d'Educació Especial Jeroni de Moragas i C. Torras Assistents: Tots els convocats Convoca: C. Torras Secretari: C. Torras	Lloc, data i hora: CEE, Mora d'Ebre, 18/02/15, 17:15-18:30h	

- En relació a conceptes d'IT, es fa palès la manca de coneixement per part dels professionals del CEE de diversos termes i serveis. Així, CT clarifica diversos punts:
 - Distinció entre dades i arxius.
 - Distinció entre sistema operatiu i programes.
 - Concepte d'interfície. Diferència entre programes locals i interfícies web entre l'usuari i la base de dades.
- Els professionals del CEE no disposen de cap servei de dades. Funcionen completament només a través d'arxius. Per exemple, els llistats d'alumnes són a través d'arxius de processadors de text. Durant la conversa, es fa palès el poc coneixement de la potencialitat de treballar amb dades que tenen els professionals.
- Els professionals comprenen la situació caòtica actual, basada en què cada ordinador hi ha programes diferents, i això els ocasiona constants problemes de compatibilitat, multimèdia que no es pot obrir, etc. CT insisteix que el fet de disposar d'una política de centre d'IT i que tots els ordinadors tinguin el mateix programari eliminarà gairebé tots aquests problemes.
- Els treballadors comenten la utilitat que els seria poder fer certes tasques des de casa, la qual cosa queda anotada. CT comenta que la idea és que tots els serveis que es creïn sigui via web, per poder facilitar aquesta possibilitat.
- Els treballadors manifesten la seva preocupació per haver de canviar de programari. En aquest punt, CT explica que la gestió d'aquest canvi també forma part del projecte. S'explica que el projecte contempla tres grans fases: la de planificació en la que es definirà el programari i serveis a desenvolupar i implementar (treball conjunt amb el personal del centre), desenvolupament (treball sobretot de CT) i implementació. Aquesta darrera etapa inclou formació per part de CT als professionals del centre.
- Els treballadors comenten que el servei de correu electrònic el tenen via XTEC i que, d'entrada, es prefereix continuar fent-lo servir. En aquest cas, no requeria intervenció per part de CT.
- Es discuteix la possibilitat de tenir un bloc. Hi ha bastant consens que si es crea, es faci també a través d'XTEC, amb la qual cosa, la intervenció de CT, en tot cas, seria només de suport per crear-lo. Així, la majoria del projecte quedaria delimitat a serveis interns.
- El CEE forma part d'un ens superior: l'AJEM. Es comparteix edifici i recursos. Entre ells, el servei de xarxa. L'abast d'aquest projecte és només de CEE i per tant, cal intentar que totes les actuacions que es facin siguin independents a la resta d'infraestructura i serveis d'IT d'AJEM. Així, el proper pas és tenir una reunió específica amb el gerent d'AJEM per clarificar la situació, conèixer bé la disponibilitat de serveis, etc.
- Els treballadors del CEE comenten la necessitat de mantenir còpies de seguretat de la informació, tot i que el CEE té contractat un servei extern amb aquest objectiu. Tanmateix, tenen una manca d'eficàcia ja que estant tots acostumats a treballar en local (desar els arxius en els respectius ordinadors) i no en una carpeta de xarxa des d'on es fan les còpies de seguretat. CT explica que en el procés de creació dels serveis de dades interns, la idea és crear i posar en marxa un servidor intern que faciliti treballar al màxim contra el servidor i no en local. És a dir, que els ordinadors de cada classe es converteixin en una simple interfície. Això presenta diverses avantatges, com ara:
 - Que les dades es desin per defecte en el servidor i per tant, es doni sentit al sistema de còpies de seguretat ja implementat.
 - Que tothom tingui accés a les seves dades amb independència de l'ordinador que usi, i que a més, contindran tots els mateixos programes. Això també es pot fer extensible a ordinadors fora del CEE.

	ACTA DE REUNIÓ – PROJECTE IRENE	AR001_180215_IRENE	
	Reunió: Reunió d'inici de projecte Convocats: Professionals Centre d'Educació Especial Jeroni de Moragas i C. Torras Assistents: Tots els convocats Convoca: C. Torras Secretari: C. Torras	Lloc, data i hora: CEE, Mora d'Ebre, 18/02/15, 17:15-18:30h	

- Els treballadors pregunten que després de la implementació, la posada en marxa de tot el projectat i la finalització de la tasca de CT, quin manteniment caldrà. CT explica que es dissenyarà de manera que tot pugui funcionar de manera autònoma i que no es crearan masses més tasques de manteniment que les que existeixen actualment. S'explica que tot el programari nou, malgrat ser lliure, serà estàndard. No es preveu que el manteniment requereixi de més recursos que actualment. Al contrari, es pot esperar que la implementació de nou programari sigui més robust i personalitzat, de manera que es produeixin menys fallades. I en tot cas, el fet que tots els ordinadors disposin del mateix programari, facilitarà el manteniment en comparació a l'estat actual, en el qual cada ordinador té programari diferent.

CT explica que en tots els casos i malgrat tot, sempre es necessitarà la col·laboració de tot el personal, tant en tolerar i predisposar-se als canvis que calgui realitzar, en adoptar hàbits adequats que es recomanaran i en ajudar a identificar aquells serveis necessaris.


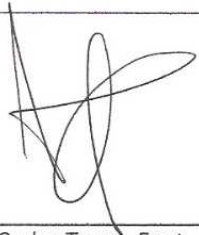
2.- Precs i preguntes



No n'hi ha.

Sense cap més tema a tractar, es dona per finalitzada la reunió a les 18:30h.

Acords

- Informar més detalladament al gerent d'AJEM del projecte i convocar una reunió amb ell per involucrar-lo en la planificació (directora del CEE).
- Identificar serveis informàtics necessaris (tots els professionals del CEE).
- Preparar la següent reunió de planificació, en la que es mostraran exemples real (C. Torras).

70 		
Anna Maria Valls Sastre		Carles Torras Font

	ACTA DE REUNIÓ – PROJECTE IRENE	AR002_270215_IRENE	
	Reunió: Reunió de planificació de projecte 1 Convocats: Josep M ^a Garcia (gerent AJEM) i Carles Torras Assistents: Josep M ^a Garcia i Carles Torras Convoca: C. Torras Secretari: C. Torras	Lloc, data i hora: AJEM, Mora d'Ebre, 27/02/15, 15:30-17:40h	

Ordre del dia

- 1.- Presentació del projecte a gerència de l'Associació (45 min).
- 2.- Anàlisi de la situació actual a l'AJEM i CEE (1 h 15 min).
- 3.- Precs i preguntes (10 min).

Acta de la reunió

S'inicia la reunió a les 15:30h de la tarda. L'objectiu de la reunió és la presentació del projecte de TFM al gerent de l'AJEM (Josep M^a Garcia) per part de C. Torras i que en Josep M^a expliqui a C. Torras la situació actual d'IT de l'Associació i els seus antecedents.

1.- Presentació del projecte a gerència de l'Associació

CT explica el context en el qual fa aquesta estada de pràctiques al CEE. Comenta que està estudiant un màster en Programari Lliure (PL) a la Universitat Oberta de Catalunya (UOC) i que una de les assignatures és el Projecte Final de Màster, el qual es pot cursar a partir d'un projecte suggerit per la UOC o cercat pel propi alumne. En aquest cas, donat que CT tenia certa coneixença del CEE i la seva situació, va proposar als responsables del CEE aquest projecte.



A continuació, CT explica resumidament en què consisteix el PL. Explica les principals diferències amb el programari propietari, avantatges i inconvenients, i en quina situació es troba en l'actualitat.

Seguidament, explica quina és la idea del projecte. La situació actual al CEE és que no existeix ni ha existit mai una estratègia de Tecnologies d'Informació (IT). Fins ara, el CEE s'ha alimentat d'ordinadors cedits per diverses entitats i algun puntualment comprat. JG puntualitza que això va ser així als principis de l'Associació, però que ens els darrers anys, la majoria d'entrada de nous ordinadors ja és mitjançant compra, i que d'aquesta manera, el parc actual d'ordinadors està prou actualitzat.

Es reconeix que cada ordinador disposa d'un programari determinat, diferent en cadascun, sense cap política en quan als propis programes, seguretat, etc. Així mateix, el CEE no disposa de cap servei basat en gestió de dades. A partir d'aquí, es proposa dissenyar una estratègia d'IT que contempli:

- Definició de programari oficial del CEE, basat en PL. La idea és que totes les màquines disposin del mateix programari i sigui PL. I que aquest sigui identificat a través d'un procés de pluja d'idees de necessitats d'IT dut a terme per tots els professionals del CEE.
 - El grau d'implementació de nou programari lliure vindrà determinat per l'estudi de necessitats que es realitzarà en la primera fase del projecte.
- Definició de serveis de bases de dades a desenvolupar i implementar.

JG reafirma la utilitat i conveniència que pot tenir el projecte, però bé que s'ha de tenir molt en compte la dificultat que certs usuaris poden tenir en adoptar nou programari. CT explica que el projecte d'implementació té en compte això (gestió del canvi) i que existeixen certes accions destinades a mitigar aquest punt, com són la formació i el desenvolupament de guies/manuals pels usuaris.

	ACTA DE REUNIÓ – PROJECTE IRENE	AR002_270215_IRENE	
	Reunió: Reunió de planificació de projecte 1 Convocats: Josep M ^è Garcia (gerent AJEM) i Carles Torras Assistents: Josep M ^è Garcia i Carles Torras Convoca: C. Torras Secretari: C. Torras	Lloc, data i hora: AJEM, Mora d'Ebre, 27/02/15, 15:30-17:40h	

2.- Anàlisi de la situació actual a l'AJEM i CEE

JG explica quina és la situació actual a l'AJEM en quan a IT i clarifica qüestions plantejades per CT.

En primer lloc, s'explica que AJEM disposa de connexió a Internet, proveïda per Movistar, consistent en un ample de banda de 10 MB. Disposen d'un encaminador domèstic (model Observa AW4062¹) i connexió a través d'IP dinàmica.

L'encaminador està situat a l'edifici central d'AJEM (administració), al costat d'un servidor (HP Proliant ML110 G7² amb 2 Gb de memòria RAM i dos discs durs de 250 Gb) del qual disposa l'Associació. A part, aquesta també disposa d'un SAI (Netway Technology), alguns discs durs externs i un *switch* (OvisLink EVO FSH24³). El *switch* distribueix xarxa al Taller Ocupacional per cable i al l'Escola (CEE) via una antena amplificadora (i els ordinadors del CEE disposen de dispositius receptors del senyal sense fils). Durant la reunió, es realitza una visita al lloc on hi ha aquest maquinari.

El servidor, el qual disposa com a sistema operatiu d'un Windows Server 2008 R2, només ofereix un servei que és el de còpies de seguretat de fitxers. Per això, usa el programa Cobian Backup II⁴. La metodologia implementada (a través d'un proveïdor extern) és: els usuaris treballen (desen arxius) contra una partició d'un dels dos discs durs del servidor (el servidor té 2 discs durs, un té dues particions: una pel SO i programes i l'altre per dades). En temps real, el programa de còpia sincronitza els fitxers en l'altre disc dur del servidor. I una vegada al dia (a les 24h), el programa de còpia sincronitza les dades a un tercer disc dur USB extern (*backup* incremental). Finalment i addicionalment, es disposen de 2 discs durs USB externs els quals es van intercanviant un cop a la setmana. El servidor no està connectat a Internet (només Intranet).

Donat el baix ús al qual està sotmès el servidor i el relatiu baix risc que comporta una eventual manipulació (donada la tasca que realitza i la quantitat de còpies de seguretat), es pensa en poder-lo usar per implementar els serveis que es volen desenvolupar en aquest projecte. Això, bàsicament, requereix la instal·lació d'una SGBD (PostgreSQL, MySQL: a definir) i d'un servidor web (a definir). Els avantatges són:

- Emmagatzematge de dades en una màquina interna i pròpia amb servei de còpies de seguretat.
- Màquina només disponible per intranet (simplifica els inconvenients de Protecció de Dades tot i que d'entrada no permet el treball des de casa). JG i CT estant d'acord, tanmateix, que la PD és un factor clau i prioritari. L'Associació està sotmesa al nivell més estricte en quan a PD, degut al tipus d'informació que custodien: dades personals, diagnòstics mèdics, etc. AJEM disposa d'un servei extern específic d'assessoria i control sobre PD, el qual podem usar.
- Amortització del servidor.



Es discuteix sobre quin seria un escenari final desitjable sobre l'estratègia d'IT a l'Associació. Correspondria a un xarxa 100% cablejada, on tots els ordinadors de treball d'AJEM estiguessin connectats per cable al servidor, i que aquest disposés de tots els serveis com control d'usuaris, impressió en xarxa, dades en xarxa, etc. Però actualment, tots els ordinadors del CEE es connecten via senyal sense fils amb certes deficiències que fan no aconsellable la mencionada implementació amb aquesta situació. JG comenta que es té la previsió que en un futur, es cablegi tot el centre al 100%. Llavors, es podria realitzar la implementació comentada. Així, es conclou

¹ <http://www.movistar.es/particulares/atencion-cliente/internet/adsl/equipamiento-adsl/routers/observa-aw4062/>

² http://h20564.www2.hp.com/hpsc/doc/public/display?docId=emr_na-c02793565

³ http://www.ovislink-espana.com/es/4_4int_1_18/Productos/Switches/Fast-ethernet/Evo-FSH24/

⁴ <http://www.cobiansoft.com/cobianbackup.htm>

	ACTA DE REUNIÓ – PROJECTE IRENE	AR002_270215_IRENE	
	Reunió: Reunió de planificació de projecte 1 Convocats: Josep M ^a Garcia (gerent AJEM) i Carles Torras Assistents: Josep M ^a Garcia i Carles Torras Convoca: C. Torras Secretari: C. Torras	Lloc, data i hora: AJEM, Mora d'Ebre, 27/02/15, 15:30-17:40h	

que el més adequat seria no realitzar-la ara i incloure-la com a futures tasques a desenvolupar en una segona fase dins l'actual projecte.

En relació a les màquines i programari que usen al CEE (objecte d'aquest projecte), es ratifica que no fan servir cap programa en especial que tingui requeriments, fet que facilita que es pugui implementar lliurement qualsevol SO. Tampoc no hi ha especial requeriment de maquinari, més enllà d'impressores i receptors de xarxa. CT comenta que en una primera etapa del projecte, caldrà realitzar un inventari complet dels requeriments de programari i controladors de maquinari de tot el CEE per poder realitzar una valoració final de la millor solució de programari a implementar. Durant la reunió, es realitza una visita a totes les aules del CEE per tenir una primera idea de tot el maquinari que hi ha.

JG comenta que l'AJEM té actualment adquirit el domini ajem.cat. L'usen per la pàgina web (actualment en redisseny i fase molt avançada) i pel servei de correu electrònic (que tanmateix, a nivell de CEE pràcticament no s'usa, ja que es treballa amb el d'XTEC). Tanmateix, tot això està externalitzat a una empresa de Móra d'Ebre (NUWA Agència Digital), la qual s'ocupa del disseny, i que alhora té subcontractats els serveis de CDMon per adquisició de domini i allotjament. Així, AJEM té resolt tots aquests aspectes i per tant, com ja es preveia, es mantenen fora de l'abast del present projecte.

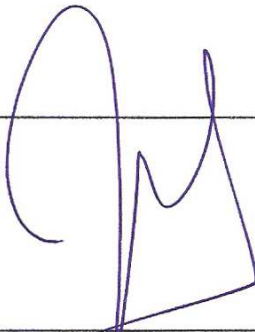
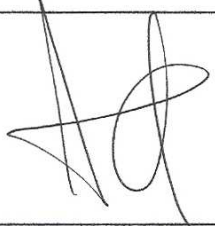
3.- Precs i preguntes (10 min).



No n'hi ha.

Abans de finalitzar la reunió, CT mostra alguns exemples a JG de SO i programari lliure i la seva funcionalitat, disseny, ergonomia, etc.

S'acorda que a partir d'aquí, CT anirà preparant la planificació del projecte (la qual ha de lliurar el dia 11 de març), a partir de tot el discutit fins ara i que s'anirà acordant entre totes les parts.

Sense cap més tema a tractar, es dóna per finalitzada la reunió a les 17:40h.

		
Josep Maria Garcia		Carles Torras Font

	ACTA DE REUNIÓ – PROJECTE IRENE	AR003_250315_IRENE	
	Reunió: Reunió de seguiment del projecte I Convocats: Professionals del CEE i Carles Torras Assistents: Professionals del CEE i Carles Torras Convoca: C. Torras Secretari: C. Torras	Lloc, data i hora: AJEM, Mora d'Ebre, 25/03/15, 17:30-19:00h	

Ordre del dia

- 1.- Presentació de la planificació (15 min).
- 2.- Presentació del bloc educatiu (30 min).
- 3.- Finalització del disseny de la base de dades del gestor de dades (30 min).
- 4.- Propers passos (15 min).

Acta de la reunió

S'inicia la reunió a les 17:30h de la tarda amb els següents professionals del CEE.

Carolina Rofes, directora del CEE en funcions	Raimunda Gris, mestre
Àfrica Ruipérez, educadora	Josep Albiac, mestre
Elisabet Solé, logopeda	Luisa Rodríguez, educadora
	Carles Torras

1.- Presentació de la planificació

CT comenta que la planificació ja va ser finalitzada i lliurada. Comenta breument les tasques que es van acordar i com s'anirien executant al llarg de la durada del projecte. S'insisteix en què a part de les tasques de desenvolupament i implementació, existeixen activitats de formació i de generació de manuals d'ús per facilitar l'adaptació a la nova situació a tots els professionals del CEE.

2.- Presentació del bloc educatiu



CT presenta i explica el nou bloc educatiu.

Primerament es comenten quines eren les premisses del disseny (simplicitat, facilitat d'ús, inclusió només de missatges i fotografies per aula, no possibilitat de comentaris i filtratge de missatges per calendari i aula) i s'explica quins motius han portat a desenvolupar una eina pròpia: personalització i control total de l'eina. S'insisteix que les bases del disseny recullen que els continguts que s'hi publicaran no seran objecte de la LOPD (continguts 100% públics). És per aquest motiu que l'eina està oberta a Internet i implementada en un servidor extern (la màquina interna d'AJEM està aïllada d'Internet per raons de seguretat).

En segon lloc, es presenta el bloc directament mostrant-ne l'ús. El bloc és actualment accessible a través de la URL blocjeroni.necessito.cat, però per entrar-hi es necessita un codi. Quan es llanci oficialment, no caldrà cap codi i es visualitzaran directament els missatges. A continuació es mostra tota la funcionalitat del bloc, com entrar missatges, eliminar-ne, filtrar, atribucions de l'administrador, etc. Es comenta que malgrat els missatges s'eliminin, la fotografia roman al servidor (així, actua com un arxiu de fotografies de seguretat).

S'explica el tractament que l'aplicació realitza amb les fotografies carregades: les hi canvia el nom afegint-hi el *timestamp* (per fer-les úniques) i les redueix a una mida de 400 px de costat (i mantenint la relació), mida apte per la publicació al bloc. S'explica que el gestor de fotografies que s'implementarà en el gestor d'informació funcionarà de manera similar (a part, s'hi podran agregar etiquetes).

Els professionals del CEE realitzen diversos comentaris. RG demana que s'afegeixi una funcionalitat: possibilitat d'incorporar un títol a la fotografia. El formulari ha de permetre poder seleccionar aquesta opció.

	ACTA DE REUNIÓ – PROJECTE IRENE	AR003_250315_IRENE	
	Reunió: Reunió de seguiment del projecte I Convocats: Professionals del CEE i Carles Torras Assistents: Professionals del CEE i Carles Torras Convoca: C. Torras Secretari: C. Torras	Lloc, data i hora: AJEM, Mora d'Ebre, 25/03/15, 17:30-19:00h	

Finalment, es comenta que s'habilitarà un codi per cada professional perquè tothom pugui anar provant l'eina. Quan estigui validada es posarà en marxa definitivament. S'acorda que no més de dues persones siguin administradores.

3.- Finalització del disseny de la base de dades del gestor de dades

Es realitza una altra iteració en el disseny de la base de dades del gestor d'informació. Es validen les entitats i relacions actuals i es realitzen algunes modificacions:

- Incorporació de categories als professionals.
- Que en les reunions puguin intervenir més d'un professional.
- Que els informes puguin signar-los més d'un professional.
- Afegir a l'entitat d'alumne: nº targeta sanitària, nom de la parella del pare, nom de la parella de la mare, nom del tutor/a.
- Que per cada incidència no només quedi registrada la data sinó també l'hora.

4.- Propers passos


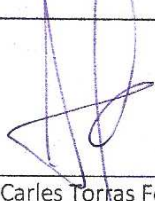
CT explica quins són els propers passos a realitzar, seguint la planificació establerta. Comenta que quan s'entri de ple en l'etapa d'implementació del nou programari, la interacció amb els professionals del centre es veurà incrementada.



CT explica que ha finalitzat l'inventari de programari/maquinari i que ja disposa de tota la informació per avaluar la implementació final del nou programari. Explica que quelcom que ha sorprès és la quantitat de dades d'usuari que hi ha a cada màquina (> 50 Gb). Els professionals comenten que són magatzems d'arxius històrics sense inventariar. CT comenta que cal que els professionals es plantegin si volen aprofitar l'ocasió per inventariar i discriminar arxius (fet que els requerirà dedicació).

En relació a l'ordinador que estava previst que estigués lliure per fer-lo servir com a banc de proves, JA comenta que l'està usant. Per tant, es fa recomanable no usar-lo per aquest propòsit. Després de conversar-ho, es decideix que els dos ordinadors que fan servir només els alumnes poden ser adequats (ni tenen dades d'usuari rellevants, ni tenen un ús crític, ni tenen programari especial, ni carpetes de xarxa, etc.).

Per acabar, CT fa una demostració de com serà el nou sistema de programari. Mostra una màquina virtual que opera amb Linux Mint i té instal·lats tota la resta de programari que s'instal·larà. En general, els professionals no hi veuen diferències molt significatives i veuen plausible l'adaptació.

Sense cap més tema a tractar, es dona per finalitzada la reunió a les 19:00h. Amb caràcter provisional, la següent reunió formal es pre-convoca pel dia 15 d'abril a les 17h.

		
Carolina Rofes		Carles Torras Font

	ACTA DE REUNIÓ – PROJECTE IRENE	AR004_220415_IRENE	
	Reunió: Reunió de seguiment del projecte II Convocats: Professionals del CEE i Carles Torras Assistents: Professionals del CEE i Carles Torras Convoca: C. Torras Secretari: C. Torras	Lloc, data i hora: AJEM, Mora d'Ebre, 22/04/15, 17:30-19:00h	

Ordre del dia

- 1.- Presentació del disseny del sistema (15 min).
- 2.- Presentació del document de política d'IT del CEE (10 min).
- 3.- Presentació de les modificacions del bloc educatiu i de la guia d'ús (30 min).
- 4.- Estat de desenvolupament del Gestor d'Informació (10 min).
- 5.- Re-planificació de la implementació del nou programari (20 min).
- 6.- Propers passos (5 min).

Acta de la reunió

S'inicia la reunió a les 17:30h de la tarda amb els següents professionals del CEE.

Carolina Rofes, directora del CEE en funcions	Raimunda Gris, mestre	Mamen
Àfrica Ruipérez, educadora	Josep Albiac, mestre	Laura Pérez, mestre
Elisabet Solé, logopeda	Luisa Rodríguez, educadora	Carles Torras

1.- Presentació del disseny del sistema

CT comenta que el disseny del sistema del projecte va ser finalitzat i lliurat. Comenta breument en què ha consistit el document i en mostra algun esquema. Amb una mica més de detall, s'explica el disseny de la base de dades del Gestor d'Informació.

2.- Presentació del document de política d'IT del CEE

CT comenta que un dels documents a generar i lliurar al CEE en el marc del projecte és el document de política d'IT. Comenta que ja ha redactat la primera versió i la presenta. Aquest document recull les directrius de programari a usar en el CEE: les necessitats que hi ha i amb quin programari es satisfan amb la corresponent justificació. Tots els ordinadors (actuals i nous) del CEE haurien de seguir les directrius presentades en la política. El contingut ja és conegut per tothom, ja que respon al què s'ha planificat i acordat conjuntament. El document és un document viu que caldrà anar-lo actualitzant en el futur d'acord amb les noves necessitats del centre i programari disponible.

3.- Presentació de les modificacions del bloc educatiu i de la guia d'ús.

CT explica que en la darrera reunió es va presentar el bloc i se'n van suggerir algunes millores. Explica que ja s'han implementat (juntament amb d'altres) i les presenta: possibilitat d'afegir títols a les fotografies, missatge d'avís abans d'eliminar un usuari, s'ha afegit un camp de descripció d'usuari i s'ha afegit la funcionalitat de llistar els usuaris actius. D'altre banda, s'han corregit altres petits errors.



Igualment, CT explica que ja ha preparat la guia d'ús del bloc. La presenta i en reparteix còpies al assistents.

Es realitzen diverses proves *in-situ* per part dels usuaris.

Finalment, s'acorda que es realitzaran unes darreres proves per part dels professionals del CEE i si no hi ha cap més novetat, es donaran instruccions a CT perquè es posi en marxa oficialment.

4.- Estat de desenvolupament del Gestor d'Informació.

CT explica que el desenvolupament del Gestor d'Informació està en un estat avançat i que properament se'n farà la presentació (tanmateix, per fer-la, és convenient que estigui relativament acabat). Explica que per provar-lo i mentre no es configura el maquinari del CEE, es desarà en una pàgina externa perquè els

	ACTA DE REUNIÓ – PROJECTE IRENE	AR004_220415_IRENE	
	Reunió: Reunió de seguiment del projecte II Convocats: Professionals del CEE i Carles Torras Assistents: Professionals del CEE i Carles Torras Convoca: C. Torras Secretari: C. Torras	Lloc, data i hora: AJEM, Mora d'Ebre, 22/04/15, 17:30-19:00h	

professionals el puguin provar (amb informació no real, i es podrà fer des de casa). Addicionalment, es prepararà també la guia d'ús.

5.- Re-planificació de la implementació del nou programari.

CT pregunta si s'ha arribat a un acord intern sobre com gestionar les dades emmagatzemades en els diversos ordinadors. CR respon que sí i que s'ha optat per bolcar directament totes les dades, donat que no es disposa temps per realitzar-ne l'inventari. Així, cada professional del CEE s'encarregarà de desar les dades del seu ordinador en un disc dur USB extern (es realitzarà dins de les properes dues setmanes). A partir d'aquí, CT proposa que per garantir les còpies, un cop s'hagi instal·lat el nou sistema en l'ordinador pilot, totes les dades es bolcaran també allà. Així, estaran a més d'un lloc. Posteriorment, un cop es vagi enllestit cadascun dels ordinadors, s'hi tornarà a bolcar la informació que hi havia i s'eliminarà de l'ordinador pilot. Així, les dades romandran en l'ordinador d'origen amb el nou sistema i en el disc dur extern, i el fet que quedi en dos llocs, és tècnicament aconsellable per cas de fallada d'algun disc dur.

Es torna a discutir sobre quin és l'ordinador més escaient per fer servir de pilot. Es confirma que un dels dos que fan servir els alumnes (com estava dissenyat) i concretament, el de l'aula Quart Creixent, el qual té menys requeriments des del punt de vista d'ús propis alumnes.

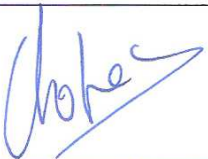

CT ratifica la importància que abans d'iniciar el procés d'implementació general, cal impartir una primera formació a tothom i haver lliurat cert material de guia. Per aquest motiu, les dates concretes de la implementació es decidiran més endavant. Es valora la possibilitat de realitzar-ho en cap de setmana (accés al CEE en cap de setmana) i completar tota la implementació o gairebé tota en un dia.



6.- Propers passos

CT explica que els propers passos, considerant que el bloc està pràcticament enllestit, són acabar la primera versió del Gestor i preparar el material de formació i guies d'ús del nou programari. Amb això fet, es podrà posar data a la primera sessió de formació, a la implantació del nou sistema a l'ordinador pilot i a la resta.

Es ressalta la necessitat de completar totes les operacions en el proper mes. Pels professionals del CEE és important fer-ho abans d'inici del període d'avaluacions (principis de juny), i CT ratifica que així és com està planificat i s'ha d'assolir, també d'acord amb el calendari del propi TFM.

Sense cap més tema a tractar, es dona per finalitzada la reunió a les 19:00h.

		
Carolina Rofes		Carles Torras

	ACTA DE REUNIÓ – PROJECTE IRENE	AR005_060515_IRENE	
	Reunió: Reunió de seguiment del projecte III + Formació Convocats: Professionals del CEE i Carles Torras Assistents: Professionals del CEE i Carles Torras Convoca: C. Torras Secretari: C. Torras	Lloc, data i hora: AJEM, Mora d'Ebre, 06/05/15, 17:00-18:30h	

Ordre del dia

- 1.- Estat del projecte (10 min).
- 2.- Presentació i formació del Gestor d'Informació (75 min).
- 3.- Propers passos (5 min).

Acta de la reunió

S'inicia la reunió a les 17:00h de la tarda amb els següents professionals del CEE.

Carolina Rofes, directora del CEE en funcions	Raimunda Gris, mestre	Mamen
Àfrica Ruipérez, educadora	Josep Albiac, mestre	Laura Pérez, mestre
Elisabet Solé, logopeda	Luisa Rodríguez, educadora	Carles Torras

1.- Estat del projecte



Es comenten en quin estat els diversos ítems del projecte:

- Bloc Educatiu. Els professionals comenten que han estat realitzant més proves i que caldria millorar uns darrers punts abans del llançament final:
 - Afegir un espaiat entre fotos d'una mateixa fila i assegurar-se que surt la barra de desplaçament horitzontal quan faci falta.
 - Que la data de les fotografies no la generi automàticament el sistema a partir de la data de càrrega, sinó que es permeti als usuaris especificar-la.
 - Possibilitat de modificació d'entrades.
- Procés de còpia de dades. En algun ordinador ja s'ha efectuat (com el de proves), però en general, encara està per fer. Es recorda que cal completar aquest procés abans de prosseguir i es recorda el calendari previst.
- Gestor. CT comenta que ja hi ha una primera versió acabada i implementada per ser provada (en servidor extern i dades fictícies). També comenta que ja hi ha una primera versió del manual apunt, el qual es reparteix. En el següent punt de l'ordre del dia es presentarà i realitzarà la primera formació.
- Nou programari. Després de comprovar idoneïtat, es decideix que el divendres 8/5 al matí, CT implementarà el nou programari en l'ordinador de proves. CT comenta que ja hi ha una primera versió del manual apunt, el qual es reparteix.

2.- Presentació i formació del Gestor d'Informació

CT presenta el Gestor i en realitza la primera formació, usant el manual ja disponible i la versió de proves. Després d'aquesta primera presentació, s'anoten les següents millores a implementar:

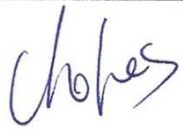

- Possibilitat d'assignar diversos tutors a una classe.
- Afegir un nom de municipi genèric.
- A la llista d'incidència ha d'aparèixer el llistat de tots els alumnes, no només els del tutor.
- Que el gestor fotogràfic també inclogui una llista d'alumnes.



	ACTA DE REUNIÓ – PROJECTE IRENE	AR005_060515_IRENE	
	Reunió: Reunió de seguiment del projecte III + Formació Convocats: Professionals del CEE i Carles Torras Assistents: Professionals del CEE i Carles Torras Convoca: C. Torras Secretari: C. Torras	Lloc, data i hora: AJEM, Mora d'Ebre, 06/05/15, 17:00-18:30h	

3.- Propers passos

S'acorda que aquest divendres 8 de maig es procedirà a la instal·lació del nou programari en l'ordinador de proves i que el proper dimecres 13 de maig es realitzarà una altra reunió/formació focalitzada en el nou programari. Entre tant, els professionals continuaran amb el procés de còpia de dades.

Sense cap més tema a tractar, es dona per finalitzada la reunió a les 18:30h.

		
Carolina Rofes		Carles Torras

	ACTA DE REUNIÓ – PROJECTE IRENE	AR006_130515_IRENE	
	Reunió: Reunió de seguiment del projecte IV + Formació Convocats: Professionals del CEE i Carles Torras Assistents: Professionals del CEE i Carles Torras Convoca: C. Torras Secretari: C. Torras	Lloc, data i hora: AJEM, Mora d'Ebre, 13/05/15, 17:00-20:00h	

Ordre del dia

- 1.- Estat del projecte (60 min).
- 2.- Presentació i formació del Nou Programari (110 min).
- 3.- Propers passos (10 min).

Acta de la reunió

S'inicia la reunió a les 17:00h de la tarda amb els següents professionals del CEE.

Carolina Rofes, directora del CEE en funcions	Raimunda Gris, mestre	Mamen
Àfrica Ruipérez, educadora	Josep Albiac, mestre	Laura Pérez, mestre
Elisabet Solé, logopeda	Luisa Rodríguez, educadora	Carles Torras



1.- Estat del projecte

Es comenten en quin estat els diversos ítems del projecte:

- **Bloc Educacional.** CT comenta i mostra els darrers canvis efectuats, segons les peticions rebudes: millora del *layout*, possibilitat de modificació de fotografies i possibilitat d'introduir una data en l'entrada dels missatges. Comenta que s'ha aprofitat l'ocasió per millorar la manera d'eliminar les entrades. Mentre es visualitza el bloc, els professionals comenten la necessitat de retocar alguna imatge. CT clarifica que això cal fer-ho amb un programa de tractament d'imatges, que en aquest cas serà el GIMP, el qual forma part del programari del CEE. En la formació, s'ensenyaran aspectes bàsics del seu ús. CT comenta que manca un darrer retoc de *layout* i que el bloc quedarà a punt per ser aprovat definitivament.
- **Gestor.** CT comenta i mostra els darrers canvis efectuats, segons les peticions rebudes: (1) possibilitat d'etiquetar fotografies amb alumnes i a partir d'un desplegable dels mateixos, (2) poder filtrar missatges també per alumne, (2) que en la creació d'incidències aparegui la llista completa d'alumnes, (3) opció de poder especificar "altres" en la llista de municipis (p.e. per alumnes estrangers) i (4) possibilitat d'assignar diversos tutors en una classe. També comenta que s'han realitzat altres ajustaments com canvi de nom d'alguns rètols, millores del *layout*, etc.
- **Nou programari.** CT comenta que el divendres 8/5 va quedar instal·lat tot el nou programari en l'ordinador pilot. Comenta una incidència que hi haver deguda a problemes amb accés a Internet, la qual va causar que algun programa quedés instal·lat en anglès. Comenta, tanmateix, que després d'aquesta la reunió, aplicarà els pedaços de català.

Els professionals comenten que han estat provant el nou programari. Les opinions són diverses. Algun professional comenta que ja en aquests moments ha començat al CEE la fase final de curs (així, abans del previst en la planificació del projecte), en la qual, el volum de feina és molt gran i demana demorar la implementació del nou programari en el seu ordinador per després d'aquesta fase escolar.

Al respecte, es realitza una discussió general per valorar les diverses opcions i replantejar la situació per tal de satisfer totes les necessitats i adaptar la planificació a les mateixes. Es duu a terme un torn d'intervencions de tots els professionals, els quals cadascun manifesten si prefereixen realitzar la implementació en el seu ordinador segons el planificat o demorar-lo. El resultat és que de les 8 màquines, en 5 s'està d'acord en realitzar-la segons el planificat i en les altres 3 posposar-ho per després de la fase de final de curs del CEE.

	ACTA DE REUNIÓ – PROJECTE IRENE	AR006_130515_IRENE	
	Reunió: Reunió de seguiment del projecte IV + Formació Convocats: Professionals del CEE i Carles Torras Assistents: Professionals del CEE i Carles Torras Convoca: C. Torras Secretari: C. Torras	Lloc, data i hora: AJEM, Mora d'Ebre, 13/05/15, 17:00-20:00h	

Finalment, es valoren dos aspectes: les necessitats del CEE i els seus professionals i les necessitats del propi projecte en quan a TFM de la UOC. Entenent que el TFM representa l'execució d'un projecte real, destinat a cobrir les necessitats del client (el CEE en aquest cas), on tenen lloc fets reals que impliquen re-planificar fites (fet que succeeix en gairebé tots els projectes), es decideix prioritzar les necessitats del CEE i en conseqüència, aprovar la re-planificació en els termes acordats a sobre.

També es considera que el nou programari quedarà instal·lat en 5 de les 8 màquines del CEE, per tant, en una majoria. Així, es valora que permet accomplir formalment amb les activitats previstes dins de la planificació del TFM.

Així, en els propers dies s'implementarà el nou programari en 5 màquines (Logopèdia, Eclipsi, Lluna Nova A, ordinador alumnes Lluna Plena i ordinador alumnes Quart Creixent). I en les 3 restats (Lluna Nova B, ordinador professional Lluna Plena i ordinador professional Quart Creixent), es realitzarà a partir de finals de juny, un cop finalitzada la fase de final de curs del CEE.

Amb els professionals implicats i la direcció es planifica el dia concret en què es realitzarà l'acció i com preparar-lo. D'acord amb el calendari establert, s'acorda que l'acció es realitzarà el dissabte 23 de maig (dia no lectiu). Per aquell dia, les dades d'aquells ordinadors caldrà que estigui desades i s'acorda el procediment per fer-ho (còpia de dades dels ordinadors a un disc dur extern i doble còpia cap a l'ordinador amb el nou programari instal·lat). Es verifiquen les carpetes dels diversos ordinadors i els disc durs disponibles.

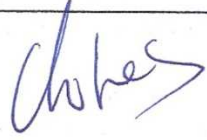

2.- Presentació i formació del Nou Programari



CT presenta el manual del nou programari i n'explica els conceptes bàsics. A partir d'aquí, es continua amb la formació usant el programari en un ordinador i el propi manual.

3.- Propers passos

S'acorda que el proper dimecres 20 de maig es continuarà amb la formació del nou programari i es ratifica el dissabte 23 de maig com a dia per realitzar la implantació de nou programari en el gruix de tots els ordinadors del CEE.

Sense cap més tema a tractar, es dona per finalitzada la reunió a les 20h.

		
Carolina Rofes		Carles Torras

	ACTA DE REUNIÓ – PROJECTE IRENE	AR007_200515_IRENE	
	Reunió: Reunió de seguiment del projecte V + Formació Convocats: Professionals del CEE i Carles Torras Assistents: Professionals del CEE i Carles Torras Convoca: C. Torras Secretari: C. Torras	Lloc, data i hora: AJEM, Mora d'Ebre, 20/05/15, 17:00-19:00h	

Ordre del dia

- 1.- Estat del projecte / aprovació del Bloc Educatiu (45 min).
- 2.- Formació del Nou Programari (60 min).
- 3.- Propers passos (15 min).

Acta de la reunió

S'inicia la reunió a les 17:00h de la tarda amb els següents professionals del CEE.

Carolina Rofes, directora del CEE en funcions	Raimunda Gris, mestre	Mamen
Àfrica Ruipérez, educadora	Josep Albiac, mestre	Laura Pérez, mestre
Elisabet Solé, logopeda	Luisa Rodríguez, educadora	Carles Torras
Josep Maria Garcia, gerent d'AJEM		



1.- Estat del projecte / aprovació del Bloc Educatiu

Es comenten en quin estat els diversos ítems del projecte:

- **Bloc Educatiu.** CT comenta i mostra els últims canvis efectuats. Amb això, s'aprova definitivament el Bloc Educatiu. En conseqüència, s'acorda realitzar-ne el llançament definitiu. Aquest consisteix en eliminar el control d'accés, mantenir els noms d'usuaris que hi ha actualment per realitzar entrades i especificar com a usuaris administradors aquells acordats en passades reunions. Finalment, sobre el contingut, s'acorda mantenir els missatges actuals nº 139, 141, 144 i 145 a fi i efecte d'activar-lo amb un mínim de contingut.
- **Gestor.** CT comenta i mostra els darrers canvis efectuats. Amb això, es pre-aprova el Gestor, a manca d'implantar-lo en el seu entorn operatiu definitiu. Això es realitzarà el dissabte 23 de maig, en l'actuació general que s'efectuarà en el maquinari del CEE. S'instal·larà en el servidor d'AJEM el programari necessari (Java i Glassfish) i s'implementarà el gestor. A partir d'aquí, caldrà iniciar el procés de càrrega de dades inicials. CT explica la metodologia que s'usarà. No s'introduiran les dades a través de l'aplicació, sinó que es faran bolcats directes sobre la BD mitjançant arxius SQL que CT prepararà/realitzarà. Això ho acomplirà a partir de la informació que els professionals li subministraran a partir de plantilles que CT els facilitarà. Aquest procés es realitzarà entre la setmana posterior a la implantació i principis de setembre per iniciar l'ús definitiu del Gestor a principis del curs vinent. El bolcat d'informació s'allargarà aquest temps degut a què manca actualitzar certa informació, hi ha l'època de final de curs al mig (molta activitat) i el període de vacances. CT recorda que és possible realitzar modificacions posteriors, altres bolcats i que diversos camps es poden anar omplint *a posteriori* (per exemple, pla de medicacions).
- **Nou programari.** El dissabte 23 de maig (dia no lectiu i de manera concentrada), CT implantarà el nou sistema a les cinc màquines acordades, tal i com estava acordat. Es planifica l'accés a les instal·lacions i subministrament de contrasenyes. Finalment, es verifica l'estat del procés de còpia de dades. Mancaven tres màquines. En dues d'elles la còpia ja s'ha efectuat i s'està realitzant la segona de seguretat. En la tercera, l'actuació està prevista pel 21 de maig.

2.- Formació del Nou Programari



Es continua amb el procés de formació usant el programari en un ordinador i la guia d'usuari creada.



	ACTA DE REUNIÓ – PROJECTE IRENE	AR007_200515_IRENE	
	Reunió: Reunió de seguiment del projecte V + Formació Convocats: Professionals del CEE i Carles Torras Assistents: Professionals del CEE i Carles Torras Convoca: C. Torras Secretari: C. Torras	Lloc, data i hora: AJEM, Mora d'Ebre, 20/05/15, 17:00-19:00h	

3.- Propers passos

A banda de les avinences explicades anteriorment, s'acorda que el proper dimecres 27 de maig es continuarà amb la formació del nou programari. CT comenta que en la mesura que els professionals tinguin disponibilitat per continuar reunint-se els dimecres a la tarda per acabar amb la formació, s'anirà completant. D'altre banda, CT ratifica que malgrat la finalització del TFM, en els propers mesos, continuarà donant el suport necessari al CEE per finalitzar la implantació del nou programari i realitzar el desplegament final del Gestor, segons el planificat.

Sense cap més tema a tractar, es dóna per finalitzada la reunió a les 19h.

		
Carolina Rofes		Carles Torras

	ACTA DE REUNIÓ – PROJECTE IRENE	AR008_270515_IRENE	
	Reunió: Reunió de seguiment del projecte VI + Formació Convocats: Professionals del CEE i Carles Torras Assistents: Professionals del CEE i Carles Torras Convoca: C. Torras Secretari: C. Torras	Lloc, data i hora: AJEM, Mora d'Ebre, 27/05/15, 17:00-19:00h	

Ordre del dia

- 1.- Estat del projecte / aprovació del Gestor (50 min).
- 2.- Formació del Nou Programari (60 min).
- 3.- Propers passos (10 min).

Acta de la reunió



S'inicia la reunió a les 17:00h de la tarda amb els següents professionals del CEE.

Carolina Rofes, directora del CEE en funcions	Raimunda Gris, mestre	Mamen
Àfrica Ruipérez, educadora	Josep Albiac, mestre	Laura Pérez, mestre
Elisabet Solé, logopeda	Luisa Rodríguez, educadora	Carles Torras

1.- Estat del projecte / aprovació del Gestor

CT comença explicant tota l'actuació realitzada el passat dissabte 23 de maig. Es va iniciar a les 10h i va finalitzar a les 01:30h (24/05). Es va treballar sobre sis màquines:

- *Servidor d'AJEM.*
 - Es va revisar l'estat del sistema i els serveis instal·lats. El sistema estava molt desactualitzat i es van aplicar actualitzacions de seguretat, de *frameworks* (.NET), etc.
 - Es va revisar l'estat del sistema de còpies *Cobian* (es va informar al gerent d'AJEM al respecte).
 - Es van instal·lar tres serveis requerits per hostatjar el nou Gestor: *Java*, *Glassfish* (servidor web) i *MySQL* (base de dades).
 - Es van implementar els arxius de la BD del Gestor i es va desplegar el Gestor pròpiament (s'ha accedeix a través de qualsevol ordinador dins la Intranet d'AJEM a través del navegador i URL: <http://192.168.1.2/>).
 - Es van detectar mancances de seguretat que es van informar al gerent d'AJEM.
- *Ordinador dels alumnes – classe Lluna Plena.*
 - Es van desar les dades dels usuaris. I després d'instal·lar el nou sistema, es van tornar a desar a la carpeta d'usuari (*usuari*: alumnes, *password*: alumnes).
 - El nou sistema amb els nous programes es van instal·lar correctament.
 - Es va reparar la unitat de DVD que no s'obria (tanmateix, està forçada i pot tornar-se a travar en qualsevol moment).
 - Quan es va iniciar l'actuació, la torre no tenia cap cable connectat (estava en el mateix estat que dimecres). Es va realitzar la instal·lació en un altre lloc i a l'anar al fer una prova al "seu lloc", es va intentar endollar en maquinari (pantalla, ratolí, etc.) i:
 - El ratolí estava totalment desmanegat (s'actua directament sobre la placa).
 - Els pins del *jack* del teclat estaven tots doblegats i girats. El teclat és inservible. Caldrà reposar-lo. Aquell era tipus PS2, però el nou pot ser USB (més robust, els pins són delicats).
 - La màquina va quedar 100% finalitzada.
- *Ordinador de l'aula Lluna Nova B.*
 - Es va realitzar una segona còpia de dades (sobre la que havia fet la Mamen) i es va repassar si estaven totes desades.

	ACTA DE REUNIÓ – PROJECTE IRENE	AR008_270515_IRENE	
	Reunió: Reunió de seguiment del projecte VI + Formació Convocats: Professionals del CEE i Carles Torras Assistents: Professionals del CEE i Carles Torras Convoca: C. Torras Secretari: C. Torras	Lloc, data i hora: AJEM, Móra d'Ebre, 27/05/15, 17:00-19:00h	

- Es va instal·lar el nou sistema i els nous programaris. Com en totes les màquines, es va crear l'usuari administrador i l'usuari habitual (en aquest cas "ana").
- Es va instal·lar la carpeta del servidor de l'usuari. Es va configurar com es va acordar. Un cop l'usuari s'autentica al sistema, a l'escriptori, se li munta automàticament la carpeta del servidor (i ja hi té accés directe sense *password*, etc.).
- Es van tornar a bolcar totes les dades de l'usuari al nou sistema (carpeta d'usuari).
- Així, aquesta màquina va quedar 100% finalitzada. Pot servir de màquina "mirall" dels professionals.
- *Ordinador dels alumnes – classe Quart Creixent.*
 - Es van realitzar ajustament pendents.
- *Ordinador de la classe Eclipsi.*
 - Ídem que l'ordinador de Lluna Nova B a excepció de la instal·lació de la carpeta del servidor de l'usuari: no es va poder realitzar per la manca de xarxa d'AJEM (pendent per quan es garanteixi l'accés a la xarxa).
- *Ordinador de la classe Logopèdia.*
 - Es va realitzar una segona còpia de dades (sobre la que havia fet l'Elisabet Solé) i es va repassar si estaven totes desades.
 - Es van fer proves amb diverses versions de *Linux Mint* amb resultats insuficients. És un ordinador molt vell (i.e. menys d'1 Gb de RAM, targeta de vídeo antiga) i els sistemes actuals, encara que siguin lleugers, no donen un rendiment suficient. Així, es va optar per deixar-hi el Windows XP que hi havia però amb tot el nou programari (*LibreOffice*, *Gimp*, etc.) com en la resta d'ordinadors.
 - Es van tornar a bolcar totes les dades de l'usuari al nou sistema (a l'escriptori, on eren).
 - Va mancar acabar un ajustament en una targeta de xarxa.
 - No es va poder instal·lar la carpeta del servidor de l'usuari per la manca de xarxa d'AJEM (pendent per quan es garanteixi l'accés a la xarxa).

Durant totes les hores, no es va tenir servei de la xarxa AJEM Escola. I la xarxa AJEM només va ser accessible a l'aula Lluna Nova B. Així, tots els accessos necessaris a Internet es van realitzar amb línies 3G.



CT comenta que cal anar pensant en substituir aquesta màquina. I en tot cas, es suggereix que com que és una d'ús d'un professional, es substitueixi per una màquina dels alumnes. Després de discutir-ho amb la direcció del CEE, s'acorda intercanviar l'ordinador dels alumnes de la classe Lluna Plena pel de Logopèdia.

També comenta que malgrat es va acordar que a totes les màquines hi hauria 2 usuaris (l'habitual i administrador), seria convenient que n'hi hagués un tercer: l'invitat. Això seria adequat pel cas d'un possible usuari que volgués utilitzar l'ordinador (és un dels avantatges del nou sistema: totes les màquines seran iguals en quan a programari) i no calgués que tingués a mà les dades de l'usuari habitual (inclosa la carpeta del servidor).

Es comenta que el Bloc Educatiu ja es va posar en marxa i s'hi fa un repàs.

CT comenta les darreres modificacions al Gestor i les mostra. Aquest queda aprovat a efectes de TFM i en espera de fer-hi el bolcat de dades inicials i iniciar-ne l'ús (previst pel setembre de 2015).

Es reparteix el nou material dels usuaris actualitzat (guia d'ús del Gestor, Bloc i nou programari).

	ACTA DE REUNIÓ – PROJECTE IRENE	AR008_270515_IRENE	
	Reunió: Reunió de seguiment del projecte VI + Formació Convocats: Professionals del CEE i Carles Torras Assistents: Professionals del CEE i Carles Torras Convoca: C. Torras Secretari: C. Torras	Lloc, data i hora: AJEM, Mora d'Ebre, 27/05/15, 17:00-19:00h	

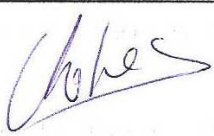
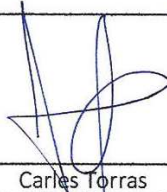
2.- Formació del Nou Programari

Es continua amb el procés de formació usant el programari en un ordinador i la guia d'usuari modificada. La formació es realitza treballant amb l'ordinador de la classe Lluna Nova B. Es repassen les maneres d'autenticar-se al sistema, els programes disponibles, etc.

3.- Propers passos



S'acorda continuar amb la formació el proper dimecres 3 de juny a la mateixa hora. Possiblement, serà la darrera sessió oficial dins del TFM, encara que després hi hagi continuïtat.

Sense cap més tema a tractar, es dona per finalitzada la reunió a les 19h.

		
Carolina Rofes		Carles Torras

Notes sobre activitats realitzades posteriorment a la reunió

- Es realitza l'intercanvi de màquina acordat (ordinador dels alumnes de la classe Lluna Plena pel de Logopèdia). Trasllet de màquines, configuració del nou ordinador de l'aula Logopèdia: xarxa, instal·lació de la carpeta de xarxa del servidor d'E. Solé, etc. Manca traslladar les dades dels alumnes al seu nou ordinador i les de l'E. Solé al seu nou ordinador. L'E. Solé se n'ocuparà.
- S'enllesteix la configuració que mancava de xarxa de l'ordinador previ de l'aula Logopèdia (convertit en la nova màquina dels alumnes de l'aula Lluna Plena).

	ACTA DE REUNIÓ – PROJECTE IRENE	AR009_030615_IRENE	
	Reunió: Reunió de seguiment del projecte V + Formació Convocats: Professionals del CEE i Carles Torras Assistents: Professionals del CEE i Carles Torras Convoca: C. Torras Secretari: C. Torras	Lloc, data i hora: AJEM, Mora d'Ebre, 03/06/15, 17:00-19:00h	

Ordre del dia

- 1.- Seguiment del projecte (20 min).
- 2.- Formació del Nou Programari (90 min).
- 3.- Propers passos (10 min).

Acta de la reunió

S'inicia la reunió a les 17:00h de la tarda amb els següents professionals del CEE.

Carolina Rofes, directora del CEE en funcions	Raimunda Gris, mestre	Mamen
Àfrica Ruipérez, educadora	Josep Albiac, mestre	Laura Pérez, mestre
Elisabet Solé, logopeda	Luisa Rodríguez, educadora	Carles Torras

1.- Estat del projecte

CT comenta que en aquests moments s'està preparant la documentació formal de finalització de projecte i que els continguts estant pràcticament enllestits.

D'altre banda, es comenta com funcionen els nous sistemes i es comparteixen dificultats, sensacions, etc.


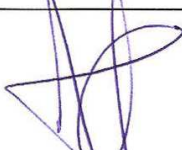
2.- Formació del Nou Programari

Es continua amb el procés de formació usant el programari en un ordinador i la guia d'usuari modificada. La formació es realitza treballant amb l'ordinador de la classe Logopèdia, connecta a la pantalla gegant. Es repassen aspectes del *LibreOffice Writer*, entre els quals, com treballar amb PDF i com generar formularis i protegir-los.

3.- Propers passos

Donats els condicionants de totes les parts (inici de l'horari reduït del CEE, finalització del projecte per part de CT, etc.), es decideix aturar temporalment les reunions. Més endavant (fora ja segurament del TFM) s'acordarà quan es continuaran (sessions de formació i procés de posada en marxa del Gestor, altres aspectes miscel·lanis).

Sense cap més tema a tractar, es dona per finalitzada la reunió a les 19h.

		
Carolina Rofes		Carles Torras

A2.2.- Informes de seguiment

El seguiment del projecte amb el tutor de la UOC s'ha realitzat mitjançant informes enviats per correu electrònic i *feedback* al respecte. També s'ha realitzat alguna trucada telefònica i una vídeo-conferència (22/05/2015, 13-14h) per revisar el funcionament de les aplicacions creades i estudiar quines millores i ajustos s'hi podia realitzar.

A continuació, es presenten els textos principals dels informes enviats.

Missatge de seguiment 1, 22/03/2015

Actualment, s'està treballant en quatre accions, d'acord amb la planificació: bloc educatiu, Sistema de Gestió de la Informació, polític del centre i inventari del maquinari.

1.- Bloc educatiu. Aquest és el punt que he treballat més fins ara, i es troba en un estat ja força avançat (és el més simple). Tinc ja una primera versió en marxa i ja he realitzat un parell de revisions amb algun professional del CEE. Està programat en Java i treballa amb *Servlets* i *AJAX*. El dimecres tinc convocada una reunió amb tots els professionals del CEE on els hi faré una presentació i espero rebre feedback general. Per llavors, més o menys, t'enviaré un enllaç perquè també li puguis donar un cop d'ull. Addicionalment, també he començat a redactar el disseny del sistema d'aquesta part. T'adjunto l'esborrany (no està revisat: és una primera redacció).

2.- Gestor. Això és el més complex. Fins ara, he treballat bastant el disseny de la base de dades, que té certa magnitud. Amb algun professional del CEE he fet un parell d'iteracions sobre la mateixa per adaptar-la bé a les seves necessitats i trobar un compromís de complexitat. I també he començat a programar la interfície. En un primer moment, pensava en usar els automatismes de *JavaFaces* per generar tot el CRUD, però al final ho he descartat per la feina posterior d'adaptació. Així, vaig programant-ho jo encara que sigui tediós. En aquí, segueixo bastant estrictament el disseny model-vista-controlador. Tot també està programat en Java i *Servlets*. En uns dies, t'adjuntaré l'esquema de la BD i els comentaris per saber-ne la teva opinió.

3.- Inventari/política del centre. Això va progressant, no tinc comentaris rellevants. Aquest dimecres tinc una altre reunió CEE i espero anar avançant en aquests temes.

Missatge de seguiment 2, 27/03/2015

Aquest dimecres vam tornar a tenir una reunió amb tot el personal del CEE. Vam repassar la planificació, els vaig presentar el bloc, vam fer una altre iteració al disseny de la base de dades i vam repassar les properes tasques. Finalment, els vaig fer ja una petita presentació de l'aspecte del *Linux Mint* i resta de programari. Adjunt, t'envio l'esborrany d'acta per la teva informació.

Així, els diversos temes estant de la següent manera:

1.- Bloc. El dono ja per bastant avançat, tot i que s'han d'anar fent retocs de peticions dels usuaris. Igualment, pot ser que vagin apareixen fallades que caldrà corregir-les. Ara està en mode de proves i accessible a tots els usuaris. Pots consultar-lo a <http://blocjeroni.necessito.cat>, t'he creat el següent codi d'usuari: XXXX Si vols entrar-hi i provar-lo, ja em diràs.

2.- Gestor. Crec que el disseny de la base de dades està ja molt avançat després d'algunes iteracions amb personal del CEE. T'adjunto un nou esborrany on ja hi ha el disseny de la base de dades. Aquesta ja comença a tenir certa complexitat. En el document s'explica tot, incloent alguna simplificació que s'ha fet. Comprenc que alguna d'aquesta simplificació seria convenient que en una versió final ja no hi fos, però per aquesta primera fase de projecte, donat el temps que hi ha, crec que és assumible. Voldria saber la teva opinió, si us plau.

Sobre el tema de la LOPD que em vas respondre, crec que la solució òptima per aquest cas és traçar l'usuari concret que es connecta, què fa (visualitza, modifica, elimina...) i tractar tota la informació com unitària. Això darrer crec que es justifica perquè realment tot va bastant en bloc (entenc que és un mini-cas comparat amb el que pugueu haver en un hospital) i ajusta la quantitat de programació als recursos disponibles.

3.- Inventari. Ja el vaig finalitzar. Està inclòs en l'esborrany que t'adjunto. Es pot dir que ara ja tinc tota la informació i estic en disposició de prendre decisions. Tanmateix, tinc dubtes, t'agrairia la teva opinió:

- Realment, hi ha una diversitat de maquinari i controladors important: diferents però propers. Ara els tinc tots recopilats i per diversos motius (que no són molt antics, que són productes prou convencionals, que la majoria són marques conegudes, que amb estàndards és suficient, que els paquets actuals de GNU/Linux estant ben servits de controladors) crec que el risc de trobar-me amb Linux Mint instal·lat i maquinari inservible per manca de controlador és baix. Veig difícil actual amb precaució màxima en el sentit de buscar la compatibilitat de cadascun dels components i recopilar tots els controladors de Windows per si de cas. Per contra, la idea (segons també el planificat) és començar per un ordinador de prova i veure com respon. Donada la proximitat entre components, això acabarà de reduir (o no) el risc per anar continuant o no.
- Un altre tema que em va sorprendre és que en gairebé cada ordinador hi ha una gran quantitat d'arxius d'usuari (> 50 Gb). Vaig parlar amb la gent i són magatzems incontrolats: històrics on ningú sap què hi ha. Vaig proposar-los de què aprofitessin l'ocasió per fer inventari + selecció, però no sé encara si ho faran. En cas que no, es pot mirar de com començar a bolcar informació indiscriminadament, però això té també cert risc i necessitat de temps "mort".

Si al final per les dues coses anteriors, el risc augmentés, es podria passar al pla de contingència de la planificació de deixar els Windows i instal·lar el PL per tota la resta segons planificat. O en els ordinador que hi hagués menys risc instal·lar el Mint i pels de més risc deixar Windows (en qualsevol cas, clar acceptant els inconvenients).

Missatge de seguiment 3, 22/04/2015

Un cop superada la fase de disseny del sistema, posem-nos al dia de com estant les coses i com procedirem.

Avui hem fet una nova reunió al CEE amb tots els professionals. Adjunto l'esborrany d'acta amb els temes tractats i acords:

- Bloc: s'han presentat les darreres novetats, s'ha realitzat alguna prova i ha quedat pràcticament beneït. Es faran unes darreres proves al CEE i si no hi ha novetat, es posarà en marxa oficialment.
- Gestor: està en fase de programació però avançada. La meua prioritat és ara acabar la primera versió complerta, penjar-la a necessito.cat i que ja tots la puguem anar provant i polint (fase de proves amb dades inventades, les dades reals només s'usaran amb la versió executant-se dins del CEE).
- Implementació del nou programari de base. Com l'acta descriu, ja hem arribat a l'acord de què fer amb les dades i com procedir. Jo ja començaré a preparar els materials de formació i guies d'usuari per iniciar la primera etapa de formació i implementació.

De moment, ja he generat el document de política d'IT del CEE (un dels *deliverables* del projecte per la banda del CEE) i la guia d'usuari del bloc. Ambdós documents els he repartit en paper avui. En aquest email els adjunto en versió electrònica.

A tots ens preocupa el tema calendari, però crec que malgrat no sobrarà res i anirà tot justet, el projecte avança a bon ritme i arribarem als terminis marcats i necessaris per tots.

Missatge de seguiment 4 (dirigit al tutor de la UOC i del CEE), 29/04/2015

He anat progressant en els diversos fronts, i aquest cap de setmana que ve, espero ja enllestir:

- Una primera versió del Gestor prou acabada i instal·lada per iniciar les proves.
- Una primera versió de documentació (incompleta) sobre el Gestor perquè ho tingueu per fer proves.
- Un primer paquet de documentació sobre el nou programari dels ordinadors.
- La caixa d'eines per venir a iniciar les obres als ordinadors prou apunt.

Amb això, caldria ja agendar específicament les properes activitats, que són:

1. Presentació del gestor.
2. 1a "formació" del nou programari de base.
3. Instal·lació del nou programari de base en l'ordinador pilot (l'ordinador dels alumnes de la classe de la Munda).

I quedaran per més endavant (cronològicament): (4) la instal·lació del programari a la resta d'ordinadors, (5) 2a formació, (6) reunions de revisió del gestor, (7) instal·lació del gestor al servidor.

Llavors, hauríem de programar ara els punt 1, 2 i 3. Cal considerar que els punts 1 i 2 requeriran cert temps de dedicació i que hi torneu a ser tots. Entenc que això ho compleixen els dimecres a la tarda, però si teniu altres opcions, diga'm-ho, si us plau. En tot cas, podem ja agendar el proper dimecres 6 de maig a la tarda? Aquesta vegada, convindria començar molt puntuals i aprofitar al màxim tot el temps. Pel punt (3), com que l'executaré sol, puc venir un altre dia en horari que no hi siguin els alumnes, que hi sigueu algú de vosaltres i després del (2): proposa'm quelcom, si us plau...

I unes altres qüestions breus:

- Com va el procés de còpia de dades?
- Pel dia que convinguem que vingui a executar el punt (3), l'ordinador en qüestió ja ha d'estar enllestit en quan a tenir totes les dades desades (tanmateix, dit de memòria, crec que en aquell ordinador hi havia molt poca cosa).
- Recorda d'avisar-me quan vulgueu activar definitivament el bloc (he vist que heu fet alguna prova més). Vam acordar que els administradors serien només 2 persones. Suposo que una tu (en quan a persona actual amb rol de directora). En tot cas, confirma'm qui seran. Diga'm també si els continguts actuals de prova cal eliminar-los (començar amb el bloc sense res) o mantenir-los.

Missatge de seguiment 5 (dirigit al tutor de la UOC i del CEE), 05/05/2015

L'execució del projecte segueix adequadament la planificació, per tant, l'estat de les diverses tasques és òptima i això ens permet encarar les darreres setmanes amb força optimisme.

- Ja hi ha la versió beta del gestor acabada i implementada per ser provada. URL: <http://jeroni.necessito.cat>
- Ja està enllestit l'esborrany de la Guia d'ús del Gestor. S'adjunta en aquest email.
- Ja està enllestit l'esborrany de la Guia d'ús del nou programari dels ordinadors del CEE. S'adjunta en aquest email.
- La guia, inclou també com fer formularis i protegir-los amb LibreOffice Writer i com importar PDF's (dos temes específics que, Carol, havíeu demanat de tenir i rebre formació de com usar-ho).
- L'equip d'intervenció (la caixa d'eines i jo) està a punt per venir a implementar el nou programari en l'ordinador pilot.

Si us plau, reviseu els dos documents i aneu provant el gestor. Per provar-lo, useu les instruccions del document adjunt. S'hi estableixen noms d'usuaris i contrasenyes per entrar, així com les dades introduïdes (fictícies) per fer proves.

Demà a les 17h, hi ha programada una nova reunió al CEE. Ordre del dia:

- Presentació del gestor i introducció al seu ús. Treballarem amb el Gestor de proves i el manual d'ús.
- Formació sobre el nou programari de base. Nova ronda de treball amb el nou programari i el manual d'ús.

- Estat del procés de còpia de dades i properes cites.

Per demà aniria bé que es tinguessin impreses algunes còpies dels dos documents. Es faran servir durant les formacions. Cal començar puntualment, ja que hi ha força feina.

Manquen 3 setmanes per lliurar la 3a PAC del TFM. Amb l'experiència de les anteriors, cal que ja hi vagi pensant. Ja m'he mirat els capítols del material i em dóna la sensació que en aquí, el contingut és més lliure. És així? Alguna recomanació? Vull evitar tenir cap sorpresa a última hora.

Missatge de seguiment 6 (dirigit al tutor de la UOC i del CEE), 08/05/2015

Aquest matí he enllestit la instal·lació del nou programari a l'ordinador pilot.

Així, l'ordinador ja disposa tant de tot el nou programari funcionant correctament, com el maquinari complementari: adaptador WIFI, impressora, etc. No hi hagut cap problema important de compatibilitat, ni de controladors, etc.

He realitzat alguns tests (incloent reproduccions de vídeo, impressions, connexions a telèfon mòbil, connexions de xarxa) i tots correctes.

Tot i que l'ordinador ja té uns quants anys, el rendiment amb el nou programari és molt bo i ràpid (ja sabem que Linux Mint incorpora components, inclosos l'interfície gràfica, poc pesants).

Prèviament, he comprovat també la documentació que hi havia en una altre compte del Windows existent, del qual havíeu perdut la contrasenya.

Ja ho aprofundider, però en general tots els ordinadors tindran dos comptes creats: un d'administrador amb drets d'administrador i un d'usuari sense drets d'administrador. Això ha garantir, entre d'altres, la robustesa del sistema. Per aquest ordinador, el nom d'usuari (normal) és alumnes i la clau alumnes (per això també crearem un protocol i una llista d'usuaris/claus). Podeu anar fent proves d'aquí a dimecres. A la Munda, li he deixat un petit informe escrit damunt la taula amb totes aquestes dades.

Finalment, una darrera petició, Carol, és que la Laura em parlava de vídeos que tenia que no podia reproduir en els sistemes dels ordinadors actuals. Caldria demanar-li que faci un recull d'aquests vídeos per comprovar si es reproduïen correctament al nou sistema.

Missatge de seguiment 7, 10/05/2015

Aquest cap de setmana he estat dedicat tot ell a programar bloc i gestor. He incorporat el següent:

- 1.- Bloc. Possibilitat d'especificar data de les entrades.
- 2.- Bloc. Possibilitat de modificar entrades.
- 3.- Bloc. Millora significativa del procés d'eliminació d'entrades.
- 4.- Bloc. Correcció de problemes de layout.
- 5.- Gestor. Possibilitat de definir diversos tutors per classe.
- 6.- Gestor. A les incidències, mostrar llista de tots els alumnes.

- 7.- Gestor. AL gestor fotogràfic, incorporar etiqueta d'alumnes a partir de la llista. Filtrar també per alumnes.
- 8.- Gestor. Correcció de problemes de layout.
- 9.- Gestor. Possibilitat de marcar usuaris com a no actius.
- 10.- Gestor. Correcció d'errors en la modificació d'items: en llistes de selecció, mostrar l'element prèviament triat.

Encara estic incorporant noves funcionalitats que m'han demanat aquesta setmana (que malauradament no es van poder identificar en el seu moment). Pel bloc, que simple, encara no està aprovat degut a això. Pel gestor, només de Java, tinc 7800 línies de codi o 479 mètodes. Més o menys, els ítems a gestionar semblen repetitius, però cada un té condicionants diferents.

Adicionalment, estic documentant-ho tot plegat i continuant amb la tasca d'implementació del nou programari a tots els ordinadors.

Missatge de seguiment 8, 25/05/2015

Aquest cap de setmana he realitzat la intervenció central al CEE. Vaig configurar el servidor, implantar el gestor i implantar tot el nou programari i configurar-lo (carpetes de xarxa, usuaris, tornar a bolcar totes les dades d'usuari, etc.) a la majoria d'ordinadors com estava previst. Més de 16 hores. Molt de temps per garantir la conservació de totes les dades (bastantes desenes de gigues), arranjar el servidor (que va bastant precari), etc. Dels ordinadors a implantar, ja estava previst que en un hi poguessin haver inconvenients (degut a la seva vellesa: 10 anys els pot tenir bons) i efectivament, cap del sistemes actuals (encara que amb escriptoris lleugers) anaven prou bé. Així, en aquesta màquina, vaig aplicar el pla de contingència: tornar a instal·lar el Windows XP que hi havia, però amb tot el programari nou com en la resta d'equips: LibreOffice, Gimp, Audacity, etc.

D'altre banda, ja estic aplicant les millores a les noves aplicacions, d'acord amb el que vam comentar a la vídeo-conferència de divendres.

Annex 3. Taules de les bases de dades

A3.1.- Taules de la base de dades del Bloc Educatiu

```
CREATE DATABASE blocjeroni DEFAULT CHARACTER SET utf8 COLLATE utf8_spanish_ci;
GRANT CREATE,INSERT,DELETE,UPDATE,SELECT on blocjeroni.* to 'jeroni'@'localhost';
SET PASSWORD FOR 'jeroni'@'localhost' = password('jeroni');
FLUSH PRIVILEGES;
```

```
CREATE TABLE aula (
    nomaula VARCHAR(30) PRIMARY KEY)
ENGINE = InnoDB;
```

```
CREATE TABLE usuari (
    clau VARCHAR(100) PRIMARY KEY,
    nomaula VARCHAR(30) NOT NULL,
    descripcio VARCHAR(100),
    admin CHAR(2) CHECK (admin IN ('si', 'no')),
    CONSTRAINT fk_nomaula
    FOREIGN KEY (nomaula)
    REFERENCES aula (nomaula))
ENGINE = InnoDB;
```

```
CREATE TABLE missatge (
    id INT AUTO_INCREMENT PRIMARY KEY,
    eltext VARCHAR(20000) NOT NULL,
    quan TIMESTAMP,
    nomaula VARCHAR(30) NOT NULL,
    CONSTRAINT fk_nomaula2
    FOREIGN KEY (nomaula)
    REFERENCES aula (nomaula))
ENGINE = InnoDB;
```

```
CREATE TABLE fotografia (
    id INT AUTO_INCREMENT PRIMARY KEY,
    fotografia VARCHAR(120) NOT NULL,
    idmissatge INT,
    titol VARCHAR(250),
    CONSTRAINT fk_idmissatge
    FOREIGN KEY (idmissatge)
    REFERENCES missatge (id))
ENGINE = InnoDB
```

A3.2.- Taules de la base de dades del Sistema de Gestió de la Informació

```

SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0;
SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS, FOREIGN_KEY_CHECKS=0;
SET @OLD_SQL_MODE=@@SQL_MODE, SQL_MODE='TRADITIONAL,ALLOW_INVALID_DATES';

CREATE SCHEMA IF NOT EXISTS `jeroni` DEFAULT CHARACTER SET utf8 COLLATE utf8_spanish_ci ;
USE `jeroni` ;

CREATE TABLE IF NOT EXISTS `jeroni`.`categoria` (
  `nomcategoria` VARCHAR(40) NOT NULL,
  PRIMARY KEY (`nomcategoria`))
ENGINE = InnoDB;

CREATE TABLE IF NOT EXISTS `jeroni`.`professional` (
  `nif` CHAR(9) NOT NULL,
  `tutoria` CHAR(2) NOT NULL,
  `lkhorari` VARCHAR(100) NULL,
  `nom` VARCHAR(45) NULL,
  `cognom1` VARCHAR(45) NULL,
  `cognom2` VARCHAR(45) NULL,
  `password` VARCHAR(100) NULL,
  `nomcategoria` VARCHAR(40) NULL,
  `lkfotousuari` VARCHAR(100) NOT NULL DEFAULT 'arxius////////documents////////fotusuaris////////generic.png',
  `actiu` CHAR(2) NOT NULL DEFAULT 'si',
  PRIMARY KEY (`nif`),
  INDEX `nomcategoria_idx` (`nomcategoria` ASC),
  CONSTRAINT `nomcategoria`
    FOREIGN KEY (`nomcategoria`)
      REFERENCES `jeroni`.`categoria` (`nomcategoria`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;

CREATE TABLE IF NOT EXISTS `jeroni`.`municipi` (
  `idmun` INT NOT NULL AUTO_INCREMENT,
  `nommunicipi` VARCHAR(150) NOT NULL,
  `nomprovincia` VARCHAR(150) NOT NULL,
  PRIMARY KEY (`idmun`))
ENGINE = InnoDB;

CREATE TABLE IF NOT EXISTS `jeroni`.`alumne` (
  `nif` CHAR(9) NOT NULL,
  `compartit` CHAR(2) NULL,
  `entrada` CHAR(4) NOT NULL,
  `naixement` DATE NULL,
  `nompare` VARCHAR(25) NULL,
  `nommare` VARCHAR(25) NULL,
  `adreca` VARCHAR(100) NULL,
  `email` VARCHAR(100) NULL,
  `telefon1` CHAR(13) NULL,
  `telefon2` CHAR(13) NULL,
  `telefon3` CHAR(13) NULL,
  `telefon4` CHAR(13) NULL,
  `idmun` INT NULL,
  `diag1` VARCHAR(500) NULL,
  `diag2` VARCHAR(500) NULL,
  `diag3` VARCHAR(500) NULL,
  `diag4` VARCHAR(500) NULL,
  `diag5` VARCHAR(500) NULL,
  `med1` VARCHAR(250) NULL,
  `med2` VARCHAR(250) NULL,
  `med3` VARCHAR(250) NULL,

```

```

`med4` VARCHAR(250) NULL,
`med5` VARCHAR(250) NULL,
`nom` VARCHAR(45) NOT NULL,
`cognom1` VARCHAR(45) NOT NULL,
`cognom2` VARCHAR(45) NOT NULL,
`lkprograma` VARCHAR(100) NULL,
`numtarsan` VARCHAR(45) NULL,
`nomparellapare` VARCHAR(25) NULL,
`nomparellamare` VARCHAR(25) NULL,
`nomtutortutora` VARCHAR(25) NULL,
PRIMARY KEY (`nif`),
INDEX `idmun_idx` (`idmun` ASC),
CONSTRAINT `idmun`
  FOREIGN KEY (`idmun`)
  REFERENCES `jeroni`.`municipi` (`idmun`)
  ON DELETE NO ACTION
  ON UPDATE NO ACTION)
ENGINE = InnoDB;

CREATE TABLE IF NOT EXISTS `jeroni`.`reunio` (
  `idreunio` INT NOT NULL AUTO_INCREMENT,
  `datahora` TIMESTAMP NOT NULL,
  `assistents` VARCHAR(500) NULL,
  `lkcontingut` VARCHAR(100) NULL,
  `propera` INT NULL,
  `alumne` CHAR(9) NOT NULL,
PRIMARY KEY (`idreunio`),
INDEX `idreunio_idx` (`propera` ASC),
INDEX `nif_idx1` (`alumne` ASC),
CONSTRAINT `idreunio`
  FOREIGN KEY (`propera`)
  REFERENCES `jeroni`.`reunio` (`idreunio`)
  ON DELETE NO ACTION
  ON UPDATE NO ACTION,
CONSTRAINT `nif9`
  FOREIGN KEY (`alumne`)
  REFERENCES `jeroni`.`alumne` (`nif`)
  ON DELETE NO ACTION
  ON UPDATE NO ACTION)
ENGINE = InnoDB;

CREATE TABLE IF NOT EXISTS `jeroni`.`informe` (
  `idinforme` INT NOT NULL AUTO_INCREMENT,
  `data` DATE NOT NULL,
  `titolinforme` VARCHAR(500) NOT NULL,
  `lkinforme` VARCHAR(100) NULL,
  `alumne` CHAR(9) NOT NULL,
PRIMARY KEY (`idinforme`),
INDEX `nif_idx1` (`alumne` ASC),
CONSTRAINT `nif6`
  FOREIGN KEY (`alumne`)
  REFERENCES `jeroni`.`alumne` (`nif`)
  ON DELETE NO ACTION
  ON UPDATE NO ACTION)
ENGINE = InnoDB;

CREATE TABLE IF NOT EXISTS `jeroni`.`aula` (
  `nomaula` VARCHAR(45) NOT NULL,
  PRIMARY KEY (`nomaula`))
ENGINE = InnoDB;

CREATE TABLE IF NOT EXISTS `jeroni`.`cursos` (
  `curs` CHAR(9) NOT NULL,
  PRIMARY KEY (`curs`))
ENGINE = InnoDB;

```

```
CREATE TABLE IF NOT EXISTS `jeroni`.`classes` (  
  `nomaula` VARCHAR(45) NOT NULL,  
  `curs` CHAR(9) NOT NULL,  
  `lkhorari` VARCHAR(100) NULL,  
  PRIMARY KEY (`nomaula`, `curs`),  
  INDEX `curs2_idx` (`curs` ASC),  
  CONSTRAINT `nomaula1`  
    FOREIGN KEY (`nomaula`)  
    REFERENCES `jeroni`.`aula` (`nomaula`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION,  
  CONSTRAINT `curs2`  
    FOREIGN KEY (`curs`)  
    REFERENCES `jeroni`.`cursos` (`curs`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION)  
ENGINE = InnoDB;  
  
CREATE TABLE IF NOT EXISTS `jeroni`.`incidencia` (  
  `idincidencia` INT NOT NULL AUTO_INCREMENT,  
  `professional` CHAR(9) NOT NULL,  
  `lkinforme` VARCHAR(100) NULL,  
  `data` TIMESTAMP NOT NULL,  
  `titol` VARCHAR(200) NOT NULL,  
  PRIMARY KEY (`idincidencia`),  
  INDEX `nif_idx` (`professional` ASC),  
  CONSTRAINT `nif7`  
    FOREIGN KEY (`professional`)  
    REFERENCES `jeroni`.`professional` (`nif`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION)  
ENGINE = InnoDB;  
  
CREATE TABLE IF NOT EXISTS `jeroni`.`incialumne` (  
  `idincidencia` INT NOT NULL,  
  `nif` CHAR(9) NOT NULL,  
  PRIMARY KEY (`idincidencia`, `nif`),  
  INDEX `nif_idx` (`nif` ASC),  
  CONSTRAINT `idincidencia`  
    FOREIGN KEY (`idincidencia`)  
    REFERENCES `jeroni`.`incidencia` (`idincidencia`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION,  
  CONSTRAINT `nif4`  
    FOREIGN KEY (`nif`)  
    REFERENCES `jeroni`.`alumne` (`nif`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION)  
ENGINE = InnoDB;  
  
CREATE TABLE IF NOT EXISTS `jeroni`.`alumclasse` (  
  `nomaula` VARCHAR(45) NOT NULL,  
  `nif` CHAR(9) NOT NULL,  
  `curs` CHAR(9) NOT NULL,  
  PRIMARY KEY (`nomaula`, `nif`, `curs`),  
  INDEX `nif_idx` (`nif` ASC),  
  INDEX `curs1_idx` (`curs` ASC),  
  CONSTRAINT `nif3`  
    FOREIGN KEY (`nif`)  
    REFERENCES `jeroni`.`alumne` (`nif`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION,  
  CONSTRAINT `nomaula2`  
    FOREIGN KEY (`nomaula`)
```

```
REFERENCES `jeroni`.`classes` (`nomaula`)
ON DELETE NO ACTION
ON UPDATE NO ACTION,
CONSTRAINT `curs1`
FOREIGN KEY (`curs`)
REFERENCES `jeroni`.`classes` (`curs`)
ON DELETE NO ACTION
ON UPDATE NO ACTION)
ENGINE = InnoDB;

CREATE TABLE IF NOT EXISTS `jeroni`.`assistencia` (
  `idassistencia` INT NOT NULL AUTO_INCREMENT,
  `data` DATE NOT NULL,
  `motiu` VARCHAR(500) NULL,
  `alumne` CHAR(9) NOT NULL,
  `professional` CHAR(9) NOT NULL,
  PRIMARY KEY (`idassistencia`),
  INDEX `nif_idx` (`professional` ASC),
  INDEX `nif_idx1` (`alumne` ASC),
  CONSTRAINT `nif10`
  FOREIGN KEY (`professional`)
  REFERENCES `jeroni`.`professional` (`nif`)
  ON DELETE NO ACTION
  ON UPDATE NO ACTION,
  CONSTRAINT `nif11`
  FOREIGN KEY (`alumne`)
  REFERENCES `jeroni`.`alumne` (`nif`)
  ON DELETE NO ACTION
  ON UPDATE NO ACTION)
ENGINE = InnoDB;

CREATE TABLE IF NOT EXISTS `jeroni`.`normes` (
  `idnormes` INT NOT NULL AUTO_INCREMENT,
  `titolnorma` VARCHAR(200) NOT NULL,
  `lknorma` VARCHAR(100) NULL,
  PRIMARY KEY (`idnormes`))
ENGINE = InnoDB;

CREATE TABLE IF NOT EXISTS `jeroni`.`profreunions` (
  `nif` CHAR(9) NOT NULL,
  `idreunio` INT NOT NULL,
  PRIMARY KEY (`nif`, `idreunio`),
  INDEX `idreunio5_idx` (`idreunio` ASC),
  CONSTRAINT `nif20`
  FOREIGN KEY (`nif`)
  REFERENCES `jeroni`.`professional` (`nif`)
  ON DELETE NO ACTION
  ON UPDATE NO ACTION,
  CONSTRAINT `idreunio5`
  FOREIGN KEY (`idreunio`)
  REFERENCES `jeroni`.`reunio` (`idreunio`)
  ON DELETE NO ACTION
  ON UPDATE NO ACTION)
ENGINE = InnoDB;

CREATE TABLE IF NOT EXISTS `jeroni`.`profinformes` (
  `nif` CHAR(9) NOT NULL,
  `idinforme` INT NOT NULL,
  PRIMARY KEY (`nif`, `idinforme`),
  INDEX `idinforme2_idx` (`idinforme` ASC),
  CONSTRAINT `nif8`
  FOREIGN KEY (`nif`)
  REFERENCES `jeroni`.`professional` (`nif`)
  ON DELETE NO ACTION
  ON UPDATE NO ACTION,
```

```
CONSTRAINT `idinforme2`
  FOREIGN KEY (`idinforme`)
  REFERENCES `jeroni`.`informe` (`idinforme`)
  ON DELETE NO ACTION
  ON UPDATE NO ACTION)
ENGINE = InnoDB;

CREATE TABLE IF NOT EXISTS `jeroni`.`fotografia` (
  `id` INT NOT NULL AUTO_INCREMENT,
  `lkfotografia` VARCHAR(100) NOT NULL,
  `nomaula` VARCHAR(45) NULL,
  `curs` CHAR(9) NULL,
  `etiqueta1` VARCHAR(45) NULL,
  `etiqueta2` VARCHAR(45) NULL,
  `etiqueta3` VARCHAR(45) NULL,
  `etiqueta4` VARCHAR(45) NULL,
  `datafot` TIMESTAMP NULL,
  PRIMARY KEY (`id`),
  INDEX `nomaula6_idx` (`nomaula` ASC),
  INDEX `curs8_idx` (`curs` ASC),
  CONSTRAINT `nomaula6`
    FOREIGN KEY (`nomaula`)
    REFERENCES `jeroni`.`classes` (`nomaula`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  CONSTRAINT `curs8`
    FOREIGN KEY (`curs`)
    REFERENCES `jeroni`.`classes` (`curs`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;

CREATE TABLE IF NOT EXISTS `jeroni`.`accio` (
  `nomaccio` VARCHAR(35) NOT NULL,
  PRIMARY KEY (`nomaccio`))
ENGINE = InnoDB;

CREATE TABLE IF NOT EXISTS `jeroni`.`acces` (
  `idacces` INT NOT NULL AUTO_INCREMENT,
  `nif` CHAR(9) NULL,
  `nomaccio` VARCHAR(35) NULL,
  `accesdh` TIMESTAMP NULL,
  PRIMARY KEY (`idacces`),
  INDEX `nomaccio1_idx` (`nomaccio` ASC),
  INDEX `nif30_idx` (`nif` ASC),
  CONSTRAINT `nomaccio1`
    FOREIGN KEY (`nomaccio`)
    REFERENCES `jeroni`.`accio` (`nomaccio`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  CONSTRAINT `nif30`
    FOREIGN KEY (`nif`)
    REFERENCES `jeroni`.`professional` (`nif`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;

CREATE TABLE IF NOT EXISTS `jeroni`.`profclasse` (
  `nifprof` CHAR(9) NOT NULL,
  `elnomaula` VARCHAR(45) NOT NULL,
  `elcursaula` CHAR(9) NOT NULL,
  PRIMARY KEY (`nifprof`, `elnomaula`, `elcursaula`),
  INDEX `lonomdaula_idx` (`elnomaula` ASC),
  INDEX `locurs_idx` (`elcursaula` ASC),
  CONSTRAINT `nifdelprof`
```

```

FOREIGN KEY (`nifprof`)
REFERENCES `jeroni`.`professional` (`nif`)
ON DELETE NO ACTION
ON UPDATE NO ACTION,
CONSTRAINT `lonomdaula`
FOREIGN KEY (`elnomaula`)
REFERENCES `jeroni`.`classes` (`nomaula`)
ON DELETE NO ACTION
ON UPDATE NO ACTION,
CONSTRAINT `locurs`
FOREIGN KEY (`elcursaula`)
REFERENCES `jeroni`.`classes` (`curs`)
ON DELETE NO ACTION
ON UPDATE NO ACTION)
ENGINE = InnoDB;

CREATE TABLE IF NOT EXISTS `jeroni`.`fotoalumne` (
  `id` INT NOT NULL,
  `nif` CHAR(9) NOT NULL,
  PRIMARY KEY (`id`, `nif`),
  INDEX `nifdelafoto_idx` (`nif` ASC),
  CONSTRAINT `iddelafoto`
    FOREIGN KEY (`id`)
    REFERENCES `jeroni`.`fotografia` (`id`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  CONSTRAINT `nifdelafoto`
    FOREIGN KEY (`nif`)
    REFERENCES `jeroni`.`alumne` (`nif`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;

CREATE TABLE IF NOT EXISTS `jeroni`.`accesals` (
  `idaccesals` INT NOT NULL,
  `nifalumne` CHAR(9) NOT NULL,
  PRIMARY KEY (`idaccesals`, `nifalumne`),
  INDEX `nifalumnes10_idx` (`nifalumne` ASC),
  CONSTRAINT `idacces10`
    FOREIGN KEY (`idaccesals`)
    REFERENCES `jeroni`.`acces` (`idacces`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  CONSTRAINT `nifalumnes10`
    FOREIGN KEY (`nifalumne`)
    REFERENCES `jeroni`.`alumne` (`nif`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;

SET SQL_MODE=@OLD_SQL_MODE;
SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS;
SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS;
USE `jeroni`;

DELIMITER $$
USE `jeroni`$$
CREATE TRIGGER `professional_BINS` BEFORE INSERT ON `professional` FOR EACH ROW
BEGIN
IF (NEW.nif REGEXP '[0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][A-Z]' ) = 0 THEN
  SIGNAL SQLSTATE '12345'
  SET MESSAGE_TEXT = 'Format incorrecte';
END IF;
END$$

```

```
USE `jeroni`$$
CREATE TRIGGER `alumne_BINS` BEFORE INSERT ON `alumne` FOR EACH ROW
BEGIN
IF (NEW.nif REGEXP '[0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][A-Z]' ) = 0 THEN
    SIGNAL SQLSTATE '12345'
        SET MESSAGE_TEXT = 'Format incorrecte';
END IF;
END$$

DELIMITER ;
```


Annex 4. Codi generat en la fase de desenvolupament

En l'execució del present projecte s'han generat dos aplicacions principals corresponents als serveis: Bloc Educatiu (T2.1) i Sistema de Gestió de la Informació (T2.2). També, s'ha generat una tercera aplicació relativa a la gestió de còpies de seguretat. En el cos de l'informe s'especifiquen els detalls de totes elles i en aquest annex es presenta el codi, el qual està organitzat per arxius i seguint l'estructura dels respectius projectes mostrada en les figures 3 i 5 respectivament.

A4.1.- Bloc Educatiu

Arxiu d'estils CSS – estilspropis.css

Descriptor – web.xml

Pàgines JSP – Carpeta “bloc”

- Elimina.jsp
- Filtra.jsp
- Gestiona.jsp
- Index.jsp
- Titols.jsp

Paquets font

- Aules.java
- Elimina.java
- Entrada.java
- Filtra.java
- Gestor.java
- Microblog.java
- Prefiltra.java
- Titol.java
- Usuaris.java
- Veureusuaris.java

Pàgina principal – index.jsp

```
<!--
Document   : Bloc educacional del CEE Jeroni de Moragas
Created on : 13/03/2015
Author    : Dr. Carles Torras Font
-->
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/functions" prefix="fn" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="utf-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="description" content="">
<meta name="author" content="Carles Torras">

<title>El bloc educacional del Jeroni de Moragas</title>

<link href="css/bootstrap.min.css" rel="stylesheet">
<link href="css/estilspropis.css" rel="stylesheet">
</head>
<body>
<jsp:forward page="./entrada"></jsp:forward>
</body>
</html>
```

Arxiu d'estils CSS – estilspropis.css

```
@CHARSET "UTF-8";

a:link {text-decoration: none; color: gray}
a:active {text-decoration: none; color: gray}
a:hover {text-decoration: none; color: black}
a:visited {text-decoration: none; color: gray}

@font-face {
  font-family: 'Slapa';
  src: url('/BlocJeroni/css/Slapa.eot?#iefix') format('embedded-opentype'),
       url('/BlocJeroni/css/Slapa.woff') format('woff'),
       url('/BlocJeroni/css/Slapa.ttf') format('truetype'),
       url('/BlocJeroni/css/Slapa.svg#Slapa') format('svg');
  font-weight: normal;
  font-style: normal;
}

body {
  font-family: 'Slapa' !important;;
  font-weight: normal;
  font-style: normal;
  font-size: 1em;
}

.capcal {
  position: fixed;
  top: 15px;
  width: 100%;
  margin-left: auto;
  margin-right: auto;
  text-align: left ;
  z-index: 10;
  font-size: 3em;
  color: #556371;
}

.cos {
  position: fixed;
  margin-top: 150px;
  z-index: 1;
  left: 0;
  right: 0;
}

.container {
  position: fixed;
  overflow-y: auto;
  overflow-x: auto;
  left: 0;
  right: 0;
}

@media screen and (max-width: 1400px) {
  .container { width: 100%; }
}

@media screen and (min-width: 1400px) and (max-width: 1600px){
  .container { width: 90%; }
}

@media screen and (min-width: 1600px) and (max-width: 1800px){
  .container { width: 80%; }
}

@media screen and (min-width: 1800px) {
  .container { width: 70%; }
}

@media screen and (max-height: 660px) {
  .container { height: 60%; }
}

@media screen and (min-height: 660px) and (max-height: 750px){
  .container { height: 70%; }
}

@media screen and (min-height: 750px) and (max-height: 900px){
  .container { height: 75%; }
}

@media screen and (min-height: 900px) {
  .container { height: 80%; }
}

.peupag {
  position: fixed;
  width: 100%;
  bottom: 0;
  height: 30px;
  background-color: #2F4F4F;
  color: white;
  z-index: 10;
  font-size: 1em;
}

.subpeupag {
  position: fixed;
  width: 100%;
  bottom: 0;
  height: 40px;
  background-color: white;
  z-index: 9;
}
```

```

}
.espaiador {
  height: 40px;
}

.navbar-custom {
  color: white;
}

.panel-custom {
  margin: 10px auto auto auto;
}

.entradamis {
  margin-top: 10px;
}

.colorfons {
  padding-top: 5px;
  padding-bottom: 5px;
  padding-left: 25px;
  padding-right: 25px;
  background-color: #E3E3E3;
}

.entradamisres {
  margin-top: 0px;
  font-size: 1.3em;
}

.llegirmis {
  margin-top: 0px;
}

.panel-heading {
  background-color: #a1bc9c!important;
}

.midapeupag {
  font-size: 1.5em;
}

.credit {
  opacity: 0.15;
}

.btn-sharp {
  border-radius:0;
  border: 0;
  background: none;
  -webkit-border-radius: 0px;
  -moz-border-radius: 0px;
}

.navbar .nav > li > a, .navbar .nav > li ,
.navbar .nav > li.current-menu-item > a,
.navbar .nav > li.current-menu-ancestor > a {
  font-size: 1.5em;
  size: bold;
}

nav ul li a:hover {
  background-color: grey !important;
}

.num_miss {
  display: inline-block;
  font-size: 1.5em;
  size: bold;
  margin-left: 100px;
  margin-top: 10px;
  color: white;
}

.nummisres {
  display: inline-block;
  font-size: 1em;
  size: bold;
  color: white;
}

```

Descriptor – web.xml

```

<?xml version="1.0" encoding="UTF-8"?>
<web-app version="3.1" xmlns="http://xmlns.jcp.org/xml/ns/javaee" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/javaee http://xmlns.jcp.org/xml/ns/javaee/web-app_3_1.xsd">
  <servlet>
    <servlet-name>microblog</servlet-name>
    <servlet-class>microblog.microblog</servlet-class>
  </servlet>
  <servlet>
    <servlet-name>gestor</servlet-name>
    <servlet-class>microblog.gestor</servlet-class>
  </servlet>
  <servlet>
    <servlet-name>usuariis</servlet-name>
    <servlet-class>microblog.usuariis</servlet-class>

```

```

</servlet>
<servlet>
  <servlet-name>elimina</servlet-name>
  <servlet-class>microblog.elimina</servlet-class>
</servlet>
<servlet>
  <servlet-name>filtra</servlet-name>
  <servlet-class>microblog.filtra</servlet-class>
</servlet>
<servlet>
  <servlet-name>prefiltra</servlet-name>
  <servlet-class>microblog.prefiltra</servlet-class>
</servlet>
<servlet>
  <servlet-name>aules</servlet-name>
  <servlet-class>microblog.aules</servlet-class>
</servlet>
<servlet>
  <servlet-name>entrada</servlet-name>
  <servlet-class>microblog.entrada</servlet-class>
</servlet>
<servlet>
  <servlet-name>titol</servlet-name>
  <servlet-class>microblog.titol</servlet-class>
</servlet>
<servlet>
  <servlet-name>veureusuari</servlet-name>
  <servlet-class>microblog.veureusuari</servlet-class>
</servlet>
<servlet-mapping>
  <servlet-name>microblog</servlet-name>
  <url-pattern>/microblog</url-pattern>
</servlet-mapping>
<servlet-mapping>
  <servlet-name>gestor</servlet-name>
  <url-pattern>/gestor</url-pattern>
</servlet-mapping>
<servlet-mapping>
  <servlet-name>usuari</servlet-name>
  <url-pattern>/usuari</url-pattern>
</servlet-mapping>
<servlet-mapping>
  <servlet-name>elimina</servlet-name>
  <url-pattern>/elimina</url-pattern>
</servlet-mapping>
<servlet-mapping>
  <servlet-name>filtra</servlet-name>
  <url-pattern>/filtra</url-pattern>
</servlet-mapping>
<servlet-mapping>
  <servlet-name>prefiltra</servlet-name>
  <url-pattern>/prefiltra</url-pattern>
</servlet-mapping>
<servlet-mapping>
  <servlet-name>aules</servlet-name>
  <url-pattern>/aules</url-pattern>
</servlet-mapping>
<servlet-mapping>
  <servlet-name>entrada</servlet-name>
  <url-pattern>/entrada</url-pattern>
</servlet-mapping>
<servlet-mapping>
  <servlet-name>titol</servlet-name>
  <url-pattern>/titol</url-pattern>
</servlet-mapping>
<servlet-mapping>
  <servlet-name>veureusuari</servlet-name>
  <url-pattern>/veureusuari</url-pattern>
</servlet-mapping>
<session-config>
  <session-timeout>
    30
  </session-timeout>
</session-config>
</web-app>

```

Elimina.jsp

```

<!--
Document   : Bloc educacional del CEE Jeroni de Moragas
Created on : 13/03/2015
Author    : Dr. Carles Torras Font
-->

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/functions" prefix="fn" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="description" content="">
    <meta name="author" content="Carles Torras">

```

```

<title>El bloc educacional del Jeroni de Moragas</title>

<!-- Bootstrap core CSS -->
<link href="css/bootstrap.min.css" rel="stylesheet">
<!-- Custom styles for this template -->
<link href="css/estilspropis.css" rel="stylesheet">

<script src="http://ajax.googleapis.com/ajax/libs/jquery/1.11.1/jquery.min.js"></script>
</head>

<body>
<div class="capcal" align='center'>
<table border="0" width="100%" cellspacing="5" cellpadding="3" align="center">
<tr><td rowspan="3" align="right"><a href="index.jsp"></a></td><td
valign="top"><b>Benvingut al Bloc Educacional</b></td></tr>
<tr><td valign="top"><b>del Jeroni de Moragas!</b></td></tr>
</table>
</div>

<div class="cos">
<div class="container">
<h3>Eliminar missatges</h3><br>
<div class="entradamis">
<form action="./elimina" class="form-horizontal" method="post" accept-charset="utf-8" id="lomeuform"
name="lomeuform">
<input type="hidden" class="form-control" id="data" name="laccio" value='elimina'>
<div class="form-group">
<label for="inputEmail3" class="col-sm-2 control-label">Usuari amb permisos per gestionar</label>
<div class="col-sm-2">
<input type="password" class="form-control" id="data" placeholder="Nom d'usuari"
name="laclau" maxlength="50" size="55">
</div>
<label for="inputEmail3" class="col-sm-2 control-label">Missatge a eliminar:</label>
<div class="col-sm-2">
<input type="text" class="form-control" id="data" name="elimina" maxlength="5" style="width:
80px" value='${numbru}'>
</div>
</div>
<button type="submit" name="submit" id="submit" value="Publicar" class="btn btn-default btn-
mini">Elimina missatge</button>
</form>
</div><br><br>

<h3>Modificar missatges</h3><br>
<div class="entradamis">
<form action="./elimina" class="form-horizontal" method="post" accept-charset="utf-8" id="lomeuform"
name="lomeuform">
<input type="hidden" class="form-control" id="data" name="laccio" value='modifica'>
<div class="form-group">
<label for="inputEmail3" class="col-sm-2 control-label">Usuari amb permisos per modificar</label>
<div class="col-sm-2">
<input type="password" class="form-control" id="data" placeholder="Nom d'usuari"
name="laclau" maxlength="50" size="55">
</div>
<label for="inputEmail3" class="col-sm-2 control-label">Missatge a modificar:</label>
<div class="col-sm-2">
<input type="text" class="form-control" id="data" name="elimina" maxlength="5" style="width:
80px" value='${numbru}'>
</div>
</div>
<div class="form-group">
<label for="inputEmail3" class="col-sm-2 control-label">Text</label>
<div class="col-sm-2">
<textarea class="form-control" rows="5" class="form-control" id="data" name="elmissatge"
maxlength="140" size="345"><input type="text" class="form-control" id="data" name="elmissatge"
value='${eltext}' /></textarea>
</div>
<label for="inputEmail3" class="col-sm-2 control-label">Data</label>
<div class="col-sm-2">
<input type="text" class="form-control" id="datepicker" name="ladata" style="width: 220px"
value='${quan}'>
</div>
</div><br>
<button type="submit" name="submit" id="submit" value="Modificar" class="btn btn-default btn-
mini">Modificar missatge</button>
</form>
<br><br><br><br><br><br><a href="index.jsp">Tornar</a>
</div>
</div>
<div class="espaiador"></div>
</div>

<div class="subpeupag"></div>
<div class="peupag">
<div class="container">
Per Carles Torras, 2015
</div>
</div>

<!-- Codi del datapicker -->
<link rel="stylesheet" href="http://code.jquery.com/ui/1.11.4/themes/smoothness/jquery-ui.css">
<script src="http://code.jquery.com/jquery-1.10.2.js"></script>
<script src="http://code.jquery.com/ui/1.11.4/jquery-ui.js"></script>
<link rel="stylesheet" href="css/jquery-ui-timepicker-addon.css">
<script type="text/javascript" src="js/jquery-ui-timepicker-addon.js"></script>
<script>
<script>
$(function() {
$( '#datepicker' ).datetimepicker({ dateFormat: "yy-mm-dd", timeFormat: "hh:mm:ss" });
});
</script>
</script>

```



```

        </label>
        <label>
          <input type="radio" name="optionsRadios2" id="optionsRadios2" value="noadmin">
            No administrador
        </label>
      </div><br><br>
      <div class="form-group">
        <label for="Select">Elegeix una aula</label>
        <select id="Select" class="form-control" name="laopcio" style="width: 300px">
          <c:forEach items="{resposta}" var="option">
            <option value="{option}">{option}</option>
          </c:forEach>
        </select>
      </div>
      <button type="submit" name="submit" id="submit" value="Publicar" class="btn btn-default
btn-mini">
        Executar gestiÃ³ d'usuari</button>
      </form>
      <br>
    </div><br><br>
    <td valign="top">
      <h3>GestiÃ³ d'aulas</h3>
      <div class="entradamis">
        <form action="/.aulas" class="form-horizontal" method="post" accept-charset="utf-8"
id="lomeuform" name="lomeuform">
          <input type="hidden" class="form-control" name="laclau" id="laclau2">
          <div class="radio">
            <label>
              <input type="radio" name="optionsRadios" id="optionsRadios1" value="afegir"
checked>
                Afegir l'aula
            </label>
            <label>
              <input type="radio" name="optionsRadios" id="optionsRadios2" value="eliminar">
                Eliminar l'aula
            </label>
          </div><br><br>
          <div class="form-group">
            <label for="inputEmail3" class="col-sm-2 control-label" style="width:90px">Aula a
afegir</label>
            <div class="col-sm-2">
              <input type="text" class="form-control" id="data" placeholder="Nova aula"
name="usuari"
                maxlength="50" size="55" style="width:200px">
            </div>
          </div>
          <div class="form-group">
            <label for="Select">Elegeix una aula a eliminar</label>
            <select id="Select" class="form-control" name="laopcio" style="width: 300px">
              <c:forEach items="{resposta}" var="option">
                <option value="{option}">{option}</option>
              </c:forEach>
            </select>
          </div>
          <button type="submit" name="submit" id="submit2" value="Publicar" class="btn btn-default
btn-mini">
            Executar gestiÃ³ d'aula</button>
          </form>
        </div>
      </td>
    </tr>
  </table>
  <a href="index.jsp">Tornar</a>
</div>
<div class="espaiador"></div>
</div>
<div class="subpeupag"></div>
<div class="peupag">
  <div class="container">
    Per Carles Torras, 2015
  </div>
</div>
</body>
</html>

```

Index.jsp

```

<!--
  Document   : Bloc educacional del CEE Jeroni de Moragas
  Created on  : 13/03/2015
  Author      : Dr. Carles Torras Font
-->
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/functions" prefix="fn" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="description" content="">
    <meta name="author" content="Carles Torras">

```

```

<title>El bloc educacional del Jeroni de Moragas</title>

<!-- Bootstrap core CSS -->
<link href="css/bootstrap.min.css" rel="stylesheet">
<!-- Custom styles for this template -->
<link href="css/estilspropis.css" rel="stylesheet">

<script src="http://ajax.googleapis.com/ajax/libs/jquery/1.11.1/jquery.min.js"></script>

<!-- Codi del mostrador d'entrades. Aquest Ã©s lo meu -->
<script type="text/javascript">
function loadContent() {
    var data = {"action": "c", "aula": "${aula}", "data1": "${data1}", "data2": "${data2}";
    data = $(this).serialize() + "&" + $.param(data);
    $.ajax({
        type: "POST",
        dataType: "json",
        url: "microblog",
        data: data,
        success: function(response) {
            if (response.error === 1) {
                alert(response.txterror);
            } else {
                var mida = response.missatges.length;
                $(".entradamisres").html(" ");
                for (var i = 0; i < mida; i++) {
                    $(".entradamisres").append(
                        "<div class=\"panel panel-default panel-custom\">" +
                        "<div class=\"panel-heading\"><h4 class=\"panel-title\">Missatge de la classe <b>" +
                        response.missatges[i].aula + "</b> enviat el " + response.missatges[i].data + " (" +
                        "<form style = 'display:inline;' action=\"./prefiltra\" method=\"post\" class=\"form-
inline\">"+
                        "<button class=\"btn btn-sharp\" name=\"boto\" value=\"eliminar\">" +
                        "<input type=\"hidden\" name=\"numbru\" value=\" " + response.missatges[i].id + "
\">" +
                        response.missatges[i].id + "</button></form>" +
                        ") + "</h4></div>" + "<div class=\"panel-body\">" +
                        response.missatges[i].missatge
                    );
                }
                var mida2 = response.missatges[i].fotografies.fotonum.length;
                var compta = 0;
                if (mida2 > 0) {
                    $(".entradamisres").append("<br> <table border=\"0\"> <tr>");
                    for (var j = 0; j < mida2; j++) {
                        compta = compta + 1;
                        $(".entradamisres").append(
                            "<td> \
                             <br> " +
                            response.missatges[i].titols.titol[j] + "<br><br></td>
");
                    }
                    if (compta === 3) {
                        $(".entradamisres").append("</tr><tr>");
                        compta = 0;
                    }
                }
                $(".entradamisres").append("</tr></table>");
                $(".entradamisres").append("</div></div><br><br><br>");
            }
            $(".nummisres").html(mida);
        }
    });
}

$(document).ready(function() {
    loadContent();
    $('#publish').click(function() {
        $.ajax({
            type: "POST",
            url: "microblog.php",
            data: "action=c&data="+$('#data').val(),
            success: function(response) {
                if (response.error == 0) {
                    loadContent();
                } else {
                    alert("Error!: "+response.txterror);
                }
            }
        });
    });
});
</script>

<!-- Script del veure/amagar -->
<script type="text/javascript">
function toggle_visibility(id) {
    var e = document.getElementById(id);
    if(e.style.display == 'block')
        e.style.display = 'none';
    else
        e.style.display = 'block';
}
</script>

<!-- Script per explicar com han acabat les operacions -->
<script type="text/javascript">
window.onload = function() {

```



```

<script>
    $(function() {
        $('#datepicker').datetimepicker({ dateFormat: "yy-mm-dd", timeFormat: "hh:mm:ss" });
    });
</script>

<!--<script src="http://ajax.googleapis.com/ajax/libs/jquery/2.1.1/jquery.min.js"></script-->
<script src="http://cdnjs.cloudflare.com/ajax/libs/twitter-bootstrap/3.2.0/js/bootstrap.min.js"></script>
<script src="js/highlight.pack.js"></script>
<script src="js/bootstrap-maxlength.min.js"></script>
<script>
    $(document).ready(function () {
        $('#input#data').maxlength({
            alwaysShow: true,
            warningClass: "label label-success",
            limitReachedClass: "label label-danger",
            separator: ' de ',
            preText: 'Has usat ',
            postText: ' carÀ cters disponibles.',
            validate: true
        });
        $('#textarea#data').maxlength({
            alwaysShow: true,
            warningClass: "label label-success",
            limitReachedClass: "label label-danger",
            separator: ' de ',
            preText: 'Has usat ',
            postText: ' carÀ cters disponibles.',
            validate: true
        });
        hljs.initHighlightingOnLoad();
    });
</script>
</body>
</html>

```

Titols.jsp

```

<%--
Document : Bloc educacional del CEE Jeroni de Moragas
Created on : 13/03/2015
Author : Dr. Carles Torras Font
--%>

<%page contentType="text/html" pageEncoding="UTF-8"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/functions" prefix="fn" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="utf-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="description" content="">
<meta name="author" content="Carles Torras">

<title>El bloc educacional del Jeroni de Moragas</title>

<!-- Bootstrap core CSS -->
<link href="css/bootstrap.min.css" rel="stylesheet">
<!-- Custom styles for this template -->
<link href="css/estilspropis.css" rel="stylesheet">
<script src="http://ajax.googleapis.com/ajax/libs/jquery/1.11.1/jquery.min.js"></script>
</head>

<body>
<div class="capcal" align="center">
<table border="0" width="100%" cellspacing="5" cellpadding="3" align="center">
<tr><td rowspan="3" align="right"><a href="index.jsp"></a></td><td
valign="top"><b>Benvingut al Bloc Educacional</b></td></tr>
<tr><td valign="top"><b>del Jeroni de Moragas!</b></td></tr>
</table>
</div>

<div class="cos">
<div class="container">
<h3>Introdueix els tÀ-tols per les fotos que has introduÀ-t:</h3><br>
<div class="entradamis" style="padding-left: 30px">
<form action="./titol" class="form-horizontal" method="post" accept-charset="utf-8" id="lomeuform"
name="lomeuform">
<input type="hidden" name="darrer" value="{darrer}">
<input type="hidden" name="numfotos" value="{numfotos}">
<c:forEach items="{fotos}" var="element">
<br><br>
</c:forEach>
<div class="form-group">
<c:forEach var="i" begin="1" end="{numfotos}">
<c:out value="TÀ-tol fotografia {i}: "/><input type="text" name="tit{i}"><br><br>
</c:forEach>
</div><br>
<div class="form-group">
<button type="submit" name="submit" id="submit" value="Publicar" class="btn btn-default btn-
mini">Afegir tÀ-tols</button>
</div>
</form>
<br><br><br><a href="index.jsp">Tornar</a>

```

```

        <br><br><br><br><br><br><br><br><br><br><br><br><br><br>
    </div>
    <div class="entradamisres"></div>
</div>
<div class="espaiador"></div>
</div>
<div class="subpeupag"></div>
<div class="peupag">
    <div class="container">
        Per Carles Torras, 2015
    </div>
</div>
</div>
</body>
</html>

```

Aules.java

```

/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package microblog;

import java.io.IOException;
import java.security.MessageDigest;
import java.security.NoSuchAlgorithmException;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;
import java.util.logging.Level;
import java.util.logging.Logger;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import java.util.ArrayList;
import java.util.List;

/**
 *
 * @author Carles Torras Font
 */
public class aules extends HttpServlet {
    private static final long serialVersionUID = 1L;

    /**
     * Processes requests for both HTTP <code>GET</code> and <code>POST</code>
     * methods.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    protected void processRequest(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        response.setContentType("text/html;charset=UTF-8");
    }

    /**
     * Handles the HTTP <code>POST</code> method.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    @Override
    protected void doPost(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {

        // Pillo dades del JSP
        String autoritzant = new String(request.getParameter("laclau").getBytes("ISO-8859-1"), "UTF-8");
        String novaaula = new String(request.getParameter("usuari").getBytes("ISO-8859-1"), "UTF-8");
        String aula = new String(request.getParameter("laopcio").getBytes("ISO-8859-1"), "UTF-8");
        String estat = request.getParameter("optionsRadios");
        String esadmin = request.getParameter("optionsRadios2");

        // Xifro el password
        StringBuffer hexString2 = new StringBuffer();
        try {
            MessageDigest md2 = MessageDigest.getInstance("SHA-256");
            md2.update(autoritzant.getBytes());
            byte byteData[] = md2.digest();
            for (int iiii=0;iiii<byteData.length;iiii++) {
                String hex2=Integer.toHexString(0xff & byteData[iiii]);
                if(hex2.length()==1) hexString2.append('0');
                hexString2.append(hex2);
            }
        } catch (NoSuchAlgorithmException ex) {
            Logger.getLogger(aules.class.getName()).log(Level.SEVERE, null, ex);
        }
        autoritzant = hexString2.toString();
    }

```

```

// Inicialitzo dades de la BD
String connexioBD = "jdbc:mysql://127.0.0.1/blocjeroni";
Connection connexio;
String con;
ResultSet rs;
String laresposta = "no";
int resultat = 0; // Codi de resposta

// Miro si l'autoritzant Ã©s correcte
try {
    Class.forName("com.mysql.jdbc.Driver");
    connexio = (Connection) DriverManager.getConnection(connexioBD, "jeroni","jeroni");
    Statement s = (Statement) connexio.createStatement();
    con = "SELECT admin FROM blocjeroni.usuari WHERE clau=" + autoritzant + " ";
    //System.out.println(con);
    rs = s.executeQuery(con); {
        while (rs.next()) {
            laresposta = rs.getString("admin");
        }
    }
} catch (ClassNotFoundException | SQLException ex) {
    Logger.getLogger(aules.class.getName()).log(Level.SEVERE, null, ex);
}

// Aplico canvis d'aula
if ("si".equals(laresposta)) {
    if ("afegir".equals(estat)) {
        try {
            Class.forName("com.mysql.jdbc.Driver");
            connexio = (Connection) DriverManager.getConnection(connexioBD, "jeroni","jeroni");
            Statement s = (Statement) connexio.createStatement();
            con = "INSERT INTO blocjeroni.aula (nomaula) VALUES (" + novaaula + " ";
            //System.out.println(con);
            s.executeUpdate(con);
            resultat = 1;
        } catch (ClassNotFoundException | SQLException ex) {
            Logger.getLogger(aules.class.getName()).log(Level.SEVERE, null, ex);
            resultat = 2;
        }
    } else {
        try {
            Class.forName("com.mysql.jdbc.Driver");
            connexio = (Connection) DriverManager.getConnection(connexioBD, "jeroni","jeroni");
            Statement s = (Statement) connexio.createStatement();
            con = "DELETE FROM blocjeroni.aula WHERE nomaula = " + aula + " ";
            //System.out.println(con);
            s.executeUpdate(con);
            resultat = 3;
        } catch (ClassNotFoundException | SQLException ex) {
            Logger.getLogger(aules.class.getName()).log(Level.SEVERE, null, ex);
            resultat = 2;
        }
    }
}

// Carrego aules
List<String> resposta = new ArrayList<String>();
try {
    Class.forName("com.mysql.jdbc.Driver");
    connexio = (Connection) DriverManager.getConnection(connexioBD, "jeroni","jeroni");
    Statement s = (Statement) connexio.createStatement();
    con = "SELECT nomaula FROM blocjeroni.aula";
    //System.out.println(con);
    rs = s.executeQuery(con); {
        while (rs.next()) {
            resposta.add(rs.getString("nomaula"));
        }
    }
} catch (ClassNotFoundException | SQLException ex) {
    Logger.getLogger(prefiltra.class.getName()).log(Level.SEVERE, null, ex);
}

request.setAttribute("resposta", resposta);
request.setAttribute("resultat", resultat);
getServletContext().getRequestDispatcher("/WEB-INF/bloc/gestiona.jsp").forward(request, response);
}
}

```

Elimina.java

```

/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package microblog;

import java.io.IOException;
import java.security.MessageDigest;
import java.security.NoSuchAlgorithmException;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;
import java.util.logging.Level;
import java.util.logging.Logger;

```

```

import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

/**
 *
 * @author Carles Torras Font
 */
public class elimina extends HttpServlet {
    private static final long serialVersionUID = 1L;

    /**
     * Processes requests for both HTTP <code>GET</code> and <code>POST</code>
     * methods.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    protected void processRequest(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        response.setContentType("text/html;charset=UTF-8");
    }

    /**
     * Handles the HTTP <code>POST</code> method.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    @Override
    protected void doPost(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {

        // Pillo dades del JSP
        String autoritzant = new String(request.getParameter("laclau").getBytes("ISO-8859-1"), "UTF-8");
        String elimina = request.getParameter("elimina");
        String accio = request.getParameter("laccio");
        //System.out.println(elimina + " " + accio);
        int resultat = 0;

        // Xifro els passwords
        StringBuffer hexString = new StringBuffer();
        try {
            MessageDigest md = MessageDigest.getInstance("SHA-256");
            md.update(autoritzant.getBytes());
            byte byteData[] = md.digest();
            for (int i=0;i<byteData.length;i++) {
                String hex=Integer.toHexString(0xff & byteData[i]);
                if(hex.length()==1) hexString.append('0');
                hexString.append(hex);
            }
        } catch (NoSuchAlgorithmException ex) {
            Logger.getLogger(elimina.class.getName()).log(Level.SEVERE, null, ex);
        }
        autoritzant = hexString.toString();

        // Inicialitzo dades de la BD
        String connexioBD = "jdbc:mysql://127.0.0.1/blocjeroni";
        Connection connexio;
        String con;
        ResultSet rs;
        String resposta = "potser";
        String resposta2 = "no";
        String resposta3 = "cap";

        // Miro si l'autoritzant Ã©s administrador i quina aula tÃ©
        try {
            Class.forName("com.mysql.jdbc.Driver");
            connexio = (Connection) DriverManager.getConnection(connexioBD, "jeroni", "jeroni");
            Statement s = (Statement) connexio.createStatement();
            con = "SELECT admin,nomaula FROM blocjeroni.usuari WHERE clau='" + autoritzant + "'";
            //System.out.println(con);
            rs = s.executeQuery (con); {
                while (rs.next()) {
                    resposta = rs.getString("admin");
                    resposta3 = rs.getString("nomaula");
                }
            }
        } catch (ClassNotFoundException | SQLException ex) {
            Logger.getLogger(elimina.class.getName()).log(Level.SEVERE, null, ex);
        }

        // Miro si l'autoritzant Ã©s l'autor
        try {
            Class.forName("com.mysql.jdbc.Driver");
            connexio = (Connection) DriverManager.getConnection(connexioBD, "jeroni", "jeroni");
            Statement s = (Statement) connexio.createStatement();
            con = "SELECT nomaula FROM blocjeroni.missatge WHERE id='" + elimina + "'";
            //System.out.println(con);
            rs = s.executeQuery (con); {
                while (rs.next()) {
                    resposta2 = rs.getString("nomaula");
                }
            }
        }
    }
}

```

```

} catch (ClassNotFoundException | SQLException ex) {
    Logger.getLogger(elimina.class.getName()).log(Level.SEVERE, null, ex);
}

// Elimino o modifoc el missatge si l'usuari és admin o l'autor
if ("si".equals(resposta)) {
    if ("elimina".equals(accio)) {
        // Primer les fotografies associades...
        try {
            Class.forName("com.mysql.jdbc.Driver");
            connexio = (Connection) DriverManager.getConnection(connexioBD, "jeroni", "jeroni");
            Statement s = (Statement) connexio.createStatement();
            con = "DELETE FROM blocjeroni.fotografia WHERE idmissatge = " + elimina + ";";
            //System.out.println(con);
            s.executeUpdate(con);
            resultat = 4;
        } catch (ClassNotFoundException | SQLException ex) {
            Logger.getLogger(elimina.class.getName()).log(Level.SEVERE, null, ex);
            resultat = 2;
        }
    }
    try {
        Class.forName("com.mysql.jdbc.Driver");
        connexio = (Connection) DriverManager.getConnection(connexioBD, "jeroni", "jeroni");
        Statement s = (Statement) connexio.createStatement();
        con = "DELETE FROM blocjeroni.missatge WHERE id = " + elimina + ";";
        //System.out.println(con);
        s.executeUpdate(con);
        resultat = 4;
    } catch (ClassNotFoundException | SQLException ex) {
        Logger.getLogger(elimina.class.getName()).log(Level.SEVERE, null, ex);
        resultat = 2;
    }
} else if ("modifica".equals(accio)) {
    String text = new String(request.getParameter("elmissatge").getBytes("ISO-8859-1"), "UTF-8");
    String ladata = new String(request.getParameter("ladata").getBytes("ISO-8859-1"), "UTF-8");
    try {
        Class.forName("com.mysql.jdbc.Driver");
        connexio = (Connection) DriverManager.getConnection(connexioBD, "jeroni", "jeroni");
        Statement s = (Statement) connexio.createStatement();
        con = "UPDATE blocjeroni.missatge SET eltext=" + text + ", quan=" + ladata + " WHERE id = " +
elimina + ";";
        //System.out.println(con);
        s.executeUpdate(con);
        resultat = 5;
    } catch (ClassNotFoundException | SQLException ex) {
        Logger.getLogger(elimina.class.getName()).log(Level.SEVERE, null, ex);
        resultat = 2;
    }
}
} else if ("no".equals(resposta) && resposta2.equals(resposta3)) {
    if ("elimina".equals(accio)) {
        // Primer les fotografies associades...
        try {
            Class.forName("com.mysql.jdbc.Driver");
            connexio = (Connection) DriverManager.getConnection(connexioBD, "jeroni", "jeroni");
            Statement s = (Statement) connexio.createStatement();
            con = "DELETE FROM blocjeroni.fotografia WHERE idmissatge = " + elimina + ";";
            //System.out.println(con);
            s.executeUpdate(con);
            resultat = 4;
        } catch (ClassNotFoundException | SQLException ex) {
            Logger.getLogger(elimina.class.getName()).log(Level.SEVERE, null, ex);
            resultat = 2;
        }
    }
    try {
        Class.forName("com.mysql.jdbc.Driver");
        connexio = (Connection) DriverManager.getConnection(connexioBD, "jeroni", "jeroni");
        Statement s = (Statement) connexio.createStatement();
        con = "DELETE FROM blocjeroni.missatge WHERE id = " + elimina + ";";
        //System.out.println(con);
        s.executeUpdate(con);
        resultat = 4;
    } catch (ClassNotFoundException | SQLException ex) {
        Logger.getLogger(elimina.class.getName()).log(Level.SEVERE, null, ex);
        resultat = 2;
    }
} else if ("modifica".equals(accio)) {
    String text = new String(request.getParameter("elmissatge").getBytes("ISO-8859-1"), "UTF-8");
    String ladata = new String(request.getParameter("ladata").getBytes("ISO-8859-1"), "UTF-8");
    try {
        Class.forName("com.mysql.jdbc.Driver");
        connexio = (Connection) DriverManager.getConnection(connexioBD, "jeroni", "jeroni");
        Statement s = (Statement) connexio.createStatement();
        con = "UPDATE blocjeroni.missatge SET eltext=" + text + ", quan=" + ladata + " WHERE id = " +
elimina + ";";
        //System.out.println(con);
        resultat = 5;
        s.executeUpdate(con);
    } catch (ClassNotFoundException | SQLException ex) {
        Logger.getLogger(elimina.class.getName()).log(Level.SEVERE, null, ex);
        resultat = 2;
    }
}
}
}

request.setAttribute("resultat", resultat);
getServletContext().getRequestDispatcher("/WEB-INF/bloc/index.jsp").forward(request, response);
}
}
}

```

Entrada.java

```

package microblog;

import java.io.IOException;
import java.security.MessageDigest;
import java.security.NoSuchAlgorithmException;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;
import java.util.logging.Level;
import java.util.logging.Logger;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

/**
 *
 * @author Carles Torras Font
 */
public class entrada extends HttpServlet {
    private static final long serialVersionUID = 1L;

    /**
     * Processes requests for both HTTP <code>GET</code> and <code>POST</code>
     * methods.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    protected void processRequest(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        response.setContentType("text/html;charset=UTF-8");
    }

    @Override
    protected void doGet(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        getServletContext().getRequestDispatcher("/WEB-INF/bloc/index.jsp").forward(request, response);
    }

    /**
     * Handles the HTTP <code>POST</code> method.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    @Override
    protected void doPost(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {

        // Pillo dades del JSP
        String autoritzant = new String(request.getParameter("laclau").getBytes("ISO-8859-1"), "UTF-8");

        // Xifro els passwords
        StringBuffer hexString = new StringBuffer();
        try {
            MessageDigest md = MessageDigest.getInstance("SHA-256");
            md.update(autoritzant.getBytes());
            byte byteData[] = md.digest();
            for (int iiii=0;iiii<byteData.length;iiii++) {
                String hex=Integer.toHexString(0xff & byteData[iiii]);
                if(hex.length()==1) hexString.append('0');
                hexString.append(hex);
            }
        } catch (NoSuchAlgorithmException ex) {
            Logger.getLogger(entrada.class.getName()).log(Level.SEVERE, null, ex);
        }
        autoritzant = hexString.toString();

        // Inicialitzo dades de la BD
        String connexioBD = "jdbc:mysql://127.0.0.1/blocjeroni";
        Connection connexio;
        String con;
        ResultSet rs;
        String resposta = "prohibit";

        // Miro si l'autoritzant Ã©s administrador
        try {
            Class.forName("com.mysql.jdbc.Driver");
            connexio = (Connection) DriverManager.getConnection(connexioBD, "jeroni", "jeroni");
            Statement s = (Statement) connexio.createStatement();
            con = "SELECT admin FROM blocjeroni.usuari WHERE clau='" + autoritzant + "'";
            //System.out.println(con);
            rs = s.executeQuery(con); {
                while (rs.next()) {
                    resposta = rs.getString("admin");
                }
            }
        }
    }
}

```

```

    } catch (ClassNotFoundException | SQLException ex) {
        Logger.getLogger(entrada.class.getName()).log(Level.SEVERE, null, ex);
    }

    // Autoritzo si Ã©s o no Ã©s administrador
    if (resposta.equals("prohibit")) {
        getServletContext().getRequestDispatcher("/index.jsp").forward(request, response);
    } else {
        getServletContext().getRequestDispatcher("/WEB-INF/bloc/index.jsp").forward(request, response);
    }
}
}
}

```

Filtra.java

```

package microblog;

import java.io.IOException;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

/**
 *
 * @author Carles Torras Font
 */
public class filtra extends HttpServlet {
    private static final long serialVersionUID = 1L;

    /**
     * Processes requests for both HTTP <code>GET</code> and <code>POST</code>
     * methods.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    protected void processRequest(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        response.setContentType("text/html;charset=UTF-8");
    }

    /**
     * Handles the HTTP <code>POST</code> method.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    @Override
    protected void doPost(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {

        String aula;
        String data1;
        String data2;
        String estat = request.getParameter("optionsRadios");
        if ("calendari".equals(estat)) {
            data1 = request.getParameter("ladata1");
            data2 = request.getParameter("ladata2");
            request.setAttribute("data1", data1);
            request.setAttribute("data2", data2);
        } else if ("aula".equals(estat)) {
            aula = new String(request.getParameter("laopcio").getBytes("ISO-8859-1"), "UTF-8");
            request.setAttribute("aula", aula);
        } else if ("tot".equals(estat)) {
            aula = new String(request.getParameter("laopcio").getBytes("ISO-8859-1"), "UTF-8");
            data1 = request.getParameter("ladata1");
            data2 = request.getParameter("ladata2");
            request.setAttribute("aula", aula);
            request.setAttribute("data1", data1);
            request.setAttribute("data2", data2);
        }
        getServletContext().getRequestDispatcher("/WEB-INF/bloc/index.jsp").forward(request, response);
    }
}

```

Gestor.java

```

/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package microblog;

import java.awt.Graphics2D;
import java.awt.image.BufferedImage;
import java.io.File;
import java.io.IOException;

```

```

import java.security.MessageDigest;
import java.security.NoSuchAlgorithmException;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.Statement;
import java.text.DateFormat;
import java.text.SimpleDateFormat;
import java.util.ArrayList;
import java.util.Date;
import java.util.List;
import java.util.logging.Level;
import java.util.logging.Logger;
import javax.imageio.ImageIO;
import javax.servlet.ServletException;
import javax.servlet.annotation.MultipartConfig;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import org.apache.commons.fileupload.FileItem;
import org.apache.commons.fileupload.disk.DiskFileItemFactory;
import org.apache.commons.fileupload.servlet.ServletFileUpload;
import org.json.JSONException;
import org.json.JSONObject;

/**
 *
 * @author Carles Torras Font
 */
@WebServlet("/upload")
@MultipartConfig
public class gestor extends HttpServlet {
    private static final long serialVersionUID = 1L;

    // location to store file uploaded
    public static String UPLOAD_DIRECTORY = "imatges/";

    // upload settings
    private static final int MEMORY_THRESHOLD = 1024 * 1024 * 3; // 3MB
    private static final int MAX_FILE_SIZE = 1024 * 1024 * 40; // 40MB
    private static final int MAX_REQUEST_SIZE = 1024 * 1024 * 50; // 50MB
    public static String fileName;

    /**
     * Processes requests for both HTTP <code>GET</code> and <code>POST</code>
     * methods.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    protected void processRequest(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        response.setContentType("text/html;charset=UTF-8");
        request.setCharacterEncoding("UTF-8");
    }

    /**
     * Handles the HTTP <code>POST</code> method.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    @Override
    protected void doPost(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {

        // Creo el timestamp
        DateFormat format = new SimpleDateFormat("yyyy-MM-dd HH:mm:ss");
        Date diahora = new Date();
        String ara = format.format(diahora);
        format = new SimpleDateFormat("yyyyMMdd_HHmms");
        diahora = new Date();
        String aran = format.format(diahora);

        // configures upload settings
        DiskFileItemFactory factory = new DiskFileItemFactory();
        // sets memory threshold - beyond which files are stored in disk
        factory.setSizeThreshold(MEMORY_THRESHOLD);
        // sets temporary location to store files
        factory.setRepository(new File(System.getProperty("java.io.tmpdir")));
        ServletFileUpload upload = new ServletFileUpload(factory);
        // sets maximum size of upload file
        upload.setFileSizeMax(MAX_FILE_SIZE);
        // sets maximum size of request (include file + form data)
        upload.setSizeMax(MAX_REQUEST_SIZE);
        // constructs the directory path to store upload file
        // this path is relative to application's directory
        String uploadPath = getServletContext().getRealPath("/") + File.separator + UPLOAD_DIRECTORY;
        // creates the directory if it does not exist
        File uploadDir = new File(uploadPath);
        if (!uploadDir.exists()) {
            uploadDir.mkdir();
        }
    }
}

```

```

String data = "";
String accio = null;
String laclau = "";
String lafoto;
String darrer = "cap";
String datafoto = null;
int numfotos = 0;
List<String> foto = new ArrayList<String>();
int resultat = 0;

try {
    // parses the request's content to extract file data
    @SuppressWarnings("unchecked")
    List<FileItem> formItems = upload.parseRequest(request);

    if (formItems != null && formItems.size() > 0) {
        // iterates over form's fields
        for (FileItem item : formItems) {
            // processes only fields that are not form fields
            //System.out.println("Item: "+item.getFieldName());
            if (!item.isFormField()) {
                //Agafo la imatge
                fileName = new File(item.getName()).getName();
                fileName = fileName.replaceAll(" ", "_");
                File storeFile = new File(getServletContext().getRealPath("") +
                    File.separator + "imatges" + File.separator + fileName);
                item.write(storeFile); // deso la imatge original

                // Retoco la imatge
                BufferedImage originalImage = ImageIO.read(storeFile);
                int type = originalImage.getType() == 0? BufferedImage.TYPE_INT_ARGB : originalImage.getType();
                BufferedImage resizeImageJpg = resizeImage(originalImage, type);
                storeFile = new File(getServletContext().getRealPath("") +
                    File.separator + "imatges" + File.separator + fileName + "_mod_" + aran + ".jpg");
                ImageIO.write(resizeImageJpg, "jpg", storeFile); // deso la imatge en format jpg, amb nom ÆfÅ°nic

                // i mida adequada
                lafoto = "imatges" + File.separator + fileName + "_mod_" + aran + ".jpg"; // agafo la ruta de la
                // imatge per desar-la a MySQL
                //lafoto = lafoto.replaceAll("\\\\", "\\\\");
                lafoto = lafoto.replaceAll("\\\\", "\\\\");
                foto.add(lafoto);
            } else {
                if ("elmissatge".equals(item.getFieldName())) {
                    data = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
                    //System.out.println(data);
                    data = data.replaceAll("'", "\\\\"); // L'apÃ²strof sinÃ³, no surt
                    // Pillo l'usuari
                } else if ("laclau".equals(item.getFieldName())) {
                    laclau = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
                    // Xifro els passwords
                    StringBuffer hexString = new StringBuffer();
                    try {
                        MessageDigest md = MessageDigest.getInstance("SHA-256");
                        md.update(laclau.getBytes());
                        byte byteData[] = md.digest();
                        for (int iii=0;iii<byteData.length;iii++) {
                            String hex=Integer.toHexString(0xff & byteData[iii]);
                            if(hex.length()==1) hexString.append('0');
                            hexString.append(hex);
                        }
                    } catch (NoSuchAlgorithmException ex) {
                        Logger.getLogger(usuaris.class.getName()).log(Level.SEVERE, null, ex);
                    }
                    laclau = hexString.toString();
                } else if ("titols".equals(item.getFieldName())) {
                    // Miro si es demanen tÃ²tols de fotos
                    accio = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
                } else if ("ladata".equals(item.getFieldName())) {
                    // Data introduïda per l'usuari
                    datafoto = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
                    if (datafoto.length()<3) datafoto = ara;
                    //System.out.println(datafoto + " " + ara);
                }
            }
        }
    }
} catch (Exception ex) {
    request.setAttribute("message",
        "Hi ha hagut un error: " + ex.getMessage());
}

String connexioBD = "jdbc:mysql://127.0.0.1/blocjeroni";
Connection connexio;
String con;
String laula = "cap";
ResultSet rs;
JSONObject sortida = new JSONObject();
int mida = data.length();

if (mida > 0) {
    try {
        Class.forName("com.mysql.jdbc.Driver");
        connexio = (Connection) DriverManager.getConnection(connexioBD, "jeroni", "jeroni");
        Statement s = (Statement) connexio.createStatement();
        con = "SELECT nomaula FROM blocjeroni.usuari where clau='"+laclau+"'";
        rs = s.executeQuery(con); {
            while (rs.next()) {
                laula = rs.getString("nomaula");
            }
        }
    }
}

```

```

    }
    if (laula.equals("cap")) {
        sortida.put("error", "1");
        sortida.put("txterror", "Usuari no existent");
    } else {
        con = "INSERT INTO blocjeroni.missatge (eltext, quan, nomaula) VALUES ('" + data + "', '" + datafoto +
        "', '" + laula + "')";
        //System.out.println(con);
        s.executeUpdate(con);
        numfotos = foto.size();
        if (numfotos > 0) {
            rs = s.executeQuery("SELECT MAX(id) AS darrer FROM blocjeroni.missatge");
            while (rs.next()) {
                darrer = rs.getString("darrer");
            }
            if (!"cap".equals(darrer)) {
                connexio.setAutoCommit(false);
                PreparedStatement ps = connexio.prepareStatement("INSERT INTO blocjeroni.fotografia (idmissatge,
fotografia) VALUES ('" +
                    darrer + "', ?)");
                for (String ins:foto){
                    ps.setObject(1, ins);
                    ps.addBatch();
                }
                ps.executeBatch();
                connexio.commit();
            }
            rs = s.executeQuery("SELECT MAX(id) AS darrer FROM blocjeroni.fotografia");
            while (rs.next()) {
                darrer = rs.getString("darrer");
            }
        }
        sortida.put("error", "0");
        sortida.put("txterror", "Comentari inserat correctament");
        resultat = 1;
    }
} catch (Exception e) {
    try {
        sortida.put("error", "1");
        sortida.put("txterror", "Error a l'inserir el comentari");
        resultat = 2;
    } catch (JSONException ex) {
        Logger.getLogger(gestor.class.getName()).log(Level.SEVERE, null, ex);
        resultat = 2;
    }
}
} else {
    try {
        sortida.put("error", "2");
        sortida.put("txterror", "Sense text al comentari");
        resultat = 3;
    } catch (JSONException ex) {
        Logger.getLogger(gestor.class.getName()).log(Level.SEVERE, null, ex);
        resultat = 3;
    }
}
}
//System.out.println("Dades: "+darrer+" "+accio+" "+numfotos);
if ("siespe".equals(accio) && numfotos > 0) {
    request.setAttribute("fotos", foto);
    //System.out.println(foto.get(0));
    request.setAttribute("numfotos", numfotos);
    request.setAttribute("darrer", darrer);
    getServletContext().getRequestDispatcher("/WEB-INF/bloc/titols.jsp").forward(request, response);
} else {
    request.setAttribute("resultat", resultat);
    getServletContext().getRequestDispatcher("/WEB-INF/bloc/index.jsp").forward(request, response);
}
}

private static BufferedImage resizeImage(BufferedImage originalImage, int type){
    int nouwidth = 400; // mida de la imatge desada
    double width = originalImage.getWidth();
    double height = originalImage.getHeight();
    Double nou = (height / width) * nouwidth;
    int nouheight = nou.intValue();
    resizedImage = new BufferedImage(nouwidth, nouheight, type);
    D g = resizedImage.createGraphics();
    .drawImage(originalImage, 0, 0, nouwidth, nouheight, null);
    .dispose();
    return resizedImage;
}
}
}

```

Microblog.java

```

package microblog;

import java.io.IOException;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.Statement;
import java.util.logging.Level;
import java.util.logging.Logger;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;

```

```

import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;

/**
 *
 * @author Carles Torras Font
 */
public class microblog extends HttpServlet {

    /**
     * Processes requests for both HTTP <code>GET</code> and <code>POST</code>
     * methods.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    protected void processRequest(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        response.setContentType("text/html;charset=UTF-8");
    }

    /**
     * Handles the HTTP <code>POST</code> method.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    @Override
    protected void doPost(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {

        // Recupero dades de filtrat, si hi s'ha
        //String aula = new String(request.getParameter("aula").getBytes("ISO-8859-1"), "UTF-8");
        String aula = request.getParameter("aula");
        String data1 = request.getParameter("data1");
        String data2 = request.getParameter("data2");
        String max = request.getParameter("maxrecords");
        int maxrecords;
        if (max == null) {
            maxrecords = 10;
        } else {
            maxrecords = Integer.parseInt(request.getParameter("maxrecords"));
        }
        //System.out.println("aula: "+aula);

        //Trio el cas on estic
        int cas = 0; // Cas 0: no hi ha filtre
        int mida1 = aula.length();
        int mida2 = data1.length();
        int mida3 = data2.length();
        String cerca = "SELECT eltext,quan,nomaula,id FROM blocjeroni.missatge ORDER BY quan DESC;";
        if (mida1 > 0) {
            maxrecords = 30;
            if (mida2 > 0) {
                if (mida3 > 0) {
                    cas = 1; // Filtrem per tot
                    cerca = "SELECT eltext,quan,nomaula,id FROM blocjeroni.missatge WHERE nomaula='"
                        + aula + "' AND quan >= '" + data1 + "' AND quan <= '" + data2 + "' ORDER BY quan DESC;";
                } else {
                    cas = 2; // Filtrem per aula i data1
                    cerca = "SELECT eltext,quan,nomaula,id FROM blocjeroni.missatge WHERE nomaula='"
                        + aula + "' AND quan >= '" + data1 + "' ORDER BY quan DESC;";
                }
            } else {
                if (mida3 > 0) {
                    cas = 3; // Filtrem per aula i data 2
                    cerca = "SELECT eltext,quan,nomaula,id FROM blocjeroni.missatge WHERE nomaula='"
                        + aula + "' AND quan <= '" + data2 + "' ORDER BY quan DESC;";
                } else {
                    cas = 4; // Filtrem per aula
                    cerca = "SELECT eltext,quan,nomaula,id FROM blocjeroni.missatge WHERE nomaula='"
                        + aula + "' ORDER BY quan DESC;";
                }
            }
        } else {
            maxrecords = 30;
            if (mida2 > 0) {
                if (mida3 > 0) {
                    cas = 5; // Filtrem per data 1 i data 2
                    cerca = "SELECT eltext,quan,nomaula,id FROM blocjeroni.missatge WHERE quan >= '"
                        + data1 + "' AND quan <= '" + data2 + "' ORDER BY quan DESC;";
                } else {
                    cas = 6; // Filtrem per data1
                    cerca = "SELECT eltext,quan,nomaula,id FROM blocjeroni.missatge WHERE quan >= '"
                        + data1 + "' ORDER BY quan DESC;";
                }
            } else {
                if (mida3 > 0) {
                    cas = 7; // Filtrem per data 2
                    cerca = "SELECT eltext,quan,nomaula,id FROM blocjeroni.missatge WHERE quan <= '"
                        + data2 + "' ORDER BY quan DESC;";
                }
            }
        }
    }
}

```

```

    }
}
//System.out.println(cerca);

// Inicialitzo dades de base de dades
String connexioBD = "jdbc:mysql://127.0.0.1/blocjeroni";
String nummissatge;
String text;
Connection connexio;
ResultSet rs;
ResultSet rs2;
JSONObject sortida = new JSONObject();
JSONObject temporal = new JSONObject();
JSONObject temporal2 = new JSONObject();
JSONObject temporal3 = new JSONObject();
JSONArray jsonArray = new JSONArray();
int comptador = 0;
int comptador2 = 0;

try {
    Class.forName("com.mysql.jdbc.Driver");
    connexio = (Connection) DriverManager.getConnection(connexioBD, "jeroni", "jeroni");
    Statement s = (Statement) connexio.createStatement();
    // Llegeixo tots els missatges
    rs = s.executeQuery(cerca); {
        while (rs.next()) {
            comptador = comptador + 1;
            temporal.put("missatge", rs.getString("eltext"));
            temporal.put("data", rs.getString("quan"));
            temporal.put("aula", rs.getString("nomaula"));
            temporal.put("id", rs.getString("id"));
            nummissatge = rs.getString("id");
            // Per cada missatge, busco les fotos associades
            Statement ss = (Statement) connexio.createStatement();
            rs2 = ss.executeQuery ("SELECT fotografia,titol FROM blocjeroni.fotografia WHERE idmissatge = " +
nummissatge + ";"); {
                while (rs2.next()) {
                    comptador2 = comptador2 + 1;
                    //System.out.println("comptador2: " + comptador2);
                    temporal2.append("fotonum", rs2.getString("fotografia"));
                    text = rs2.getString("titol");
                    if (text==null) {
                        text=" ";
                    }
                    temporal3.append("titol",text);
                }
                if (comptador2 == 0) temporal2.append("fotonum","");
                if (comptador2 == 0) temporal3.append("titol","");
                //System.out.println("temporal2: "+temporal2);
                temporal.put("fotografies",temporal2);
                temporal.put("titols",temporal3);
            }
            comptador2 = 0;
            temporal2 = new JSONObject();
            temporal3 = new JSONObject();
            jsonArray.put(temporal);
            //System.out.println("temporal: "+temporal);
            temporal = new JSONObject();
            if (comptador > maxrecords - 1) break;
        }
        sortida.put("missatges", jsonArray);
    }
} catch (Exception e) {
    try {
        sortida.put("error", "1");
        sortida.put("txterror", "Error al connectar a la base de dades");
    } catch (JSONException ex) {
        Logger.getLogger(microblog.class.getName()).log(Level.SEVERE, null, ex);
    }
}
try {
    sortida.put("txterror", "Cap error");
    sortida.put("error", "0");
} catch (JSONException ex) {
    Logger.getLogger(microblog.class.getName()).log(Level.SEVERE, null, ex);
}

//System.out.println("sortida: "+sortida);
response.setContentType("application/json");
response.getWriter().write(sortida.toString());
processRequest(request, response);
}
}

```

Prefiltra.java

```

package microblog;

import java.io.IOException;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;
import java.util.ArrayList;
import java.util.List;

```

```

import java.util.logging.Level;
import java.util.logging.Logger;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

/**
 *
 * @author Carles Torras Font
 */
public class prefiltra extends HttpServlet {
    private static final long serialVersionUID = 1L;

    /**
     * Processes requests for both HTTP <code>GET</code> and <code>POST</code>
     * methods.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    protected void processRequest(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        response.setContentType("text/html;charset=UTF-8");
    }

    /**
     * Handles the HTTP <code>POST</code> method.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    @Override
    protected void doPost(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {

        // Inicialitzo dades de la BD
        String connexioBD = "jdbc:mysql://127.0.0.1/blocjeroni";
        Connection connexio;
        String con;
        ResultSet rs;
        List<String> resposta = new ArrayList<String>();
        request.setAttribute("resposta", "");

        // Carrego aules
        try {
            Class.forName("com.mysql.jdbc.Driver");
            connexio = (Connection) DriverManager.getConnection(connexioBD, "jeroni", "jeroni");
            Statement s = (Statement) connexio.createStatement();
            con = "SELECT nomaula FROM blocjeroni.aula;";
            //System.out.println(con);
            rs = s.executeQuery(con); {
                while (rs.next()) {
                    resposta.add(rs.getString("nomaula"));
                }
            }
        } catch (ClassNotFoundException | SQLException ex) {
            Logger.getLogger(prefiltra.class.getName()).log(Level.SEVERE, null, ex);
        }
        request.setAttribute("resposta", resposta);

        // Miro quã haig de fer
        String boto = request.getParameter("boto");

        // Si eliminar:
        if ("eliminar".equals(boto)) {
            String numbru = request.getParameter("numbru");
            //System.out.println(numbru);
            try {
                Class.forName("com.mysql.jdbc.Driver");
                connexio = (Connection) DriverManager.getConnection(connexioBD, "jeroni", "jeroni");
                Statement s = (Statement) connexio.createStatement();
                // Llegeixo tots els missatges
                String cerca = "SELECT eltext,quan,nomaula FROM blocjeroni.missatge WHERE id="+numbru+";";
                rs = s.executeQuery(cerca); {
                    while (rs.next()) {
                        String eltext = rs.getString("eltext");
                        String quan = rs.getString("quan");
                        String nomaula = rs.getString("nomaula");
                        request.setAttribute("eltext", eltext);
                        request.setAttribute("quan", quan);
                        request.setAttribute("nomaula", nomaula);
                        request.setAttribute("numbru", numbru);
                    }
                }
            } catch (Exception e) {}
            getServletContext().getRequestDispatcher("/WEB-INF/bloc/elimina.jsp").forward(request, response);
        }

        // Miro si haig d'anar a gestionar o filtrar
        if ("filtrar".equals(boto)) {
            getServletContext().getRequestDispatcher("/WEB-INF/bloc/filtrar.jsp").forward(request, response);
        } else if ("gestionar".equals(boto)) {
            getServletContext().getRequestDispatcher("/WEB-INF/bloc/gestiona.jsp").forward(request, response);
        }
    }
}

```



```

}
}

```

Titol.java

```

/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package microblog;

import java.io.IOException;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.Statement;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

/**
 *
 * @author Carles Torras Font
 */
@WebServlet("/titol")
public class titol extends HttpServlet {
    private static final long serialVersionUID = 1L;

    /**
     * Processes requests for both HTTP <code>GET</code> and <code>POST</code>
     * methods.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    protected void processRequest(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        response.setContentType("text/html;charset=UTF-8");
        request.setCharacterEncoding("UTF-8");
    }

    /**
     * Handles the HTTP <code>POST</code> method.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    @Override
    protected void doPost(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {

        // Pillo dades del JSP
        int darrer = Integer.parseInt(request.getParameter("darrer"));
        int numfotos = Integer.parseInt(request.getParameter("numfotos"));

        String connexioBD = "jdbc:mysql://127.0.0.1/blocjeroni";
        Connection connexio;
        String con;
        String titol;
        String element;
        int resultat = 0;

        try {
            Class.forName("com.mysql.jdbc.Driver");
            connexio = (Connection) DriverManager.getConnection(connexioBD, "jeroni", "jeroni");
            Statement s = (Statement) connexio.createStatement();
            for (int i=0; i<numfotos; i++) {
                element = "tit"+(i+1);
                //System.out.println(element);
                titol = new String(request.getParameter(element).getBytes("ISO-8859-1"), "UTF-8");
                titol = titol.replaceAll("'", "''");
                //System.out.println(titol+" "+darrer);
                con = "UPDATE blocjeroni.fotografia SET titol='"+ titol + "' WHERE id = " + (darrer-numfotos+i+1) + " ";
                //System.out.println(con);
                s.executeUpdate(con);
                resultat = 1;
            }
        } catch (Exception e) {
            resultat = 2;
        }
        request.setAttribute("resultat", resultat);
        getServletContext().getRequestDispatcher("/WEB-INF/bloc/index.jsp").forward(request, response);
    }
}

```

Usuaris.java

```

/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates

```

```

* and open the template in the editor.
*/
package microblog;

import java.io.IOException;
import java.security.MessageDigest;
import java.security.NoSuchAlgorithmException;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;
import java.util.ArrayList;
import java.util.List;
import java.util.logging.Level;
import java.util.logging.Logger;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

/**
 *
 * @author Carles Torras Font
 */
public class usuaris extends HttpServlet {
    private static final long serialVersionUID = 1L;

    /**
     * Processes requests for both HTTP <code>GET</code> and <code>POST</code>
     * methods.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    protected void processRequest(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        response.setContentType("text/html;charset=UTF-8");
    }

    /**
     * Handles the HTTP <code>POST</code> method.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    @Override
    protected void doPost(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {

        // Pillo dades del JSP
        String autoritzant = new String(request.getParameter("laclau").getBytes("ISO-8859-1"), "UTF-8");
        String usuari = new String(request.getParameter("usuari").getBytes("ISO-8859-1"), "UTF-8");
        String aula = new String(request.getParameter("laopcio").getBytes("ISO-8859-1"), "UTF-8");
        String descripcio = new String(request.getParameter("descripcio").getBytes("ISO-8859-1"), "UTF-8");
        String estat = request.getParameter("optionsRadios");
        String esadmin = request.getParameter("optionsRadios2");
        if ("admin".equals(esadmin)) {
            esadmin = "si";
        } else {
            esadmin = "no";
        }

        // Xifro els passwords
        StringBuffer hexString = new StringBuffer();
        try {
            MessageDigest md = MessageDigest.getInstance("SHA-256");
            md.update(usuari.getBytes());
            byte byteData[] = md.digest();
            for (int i=0; i<byteData.length; i++) {
                String hex=Integer.toHexString(0xff & byteData[i]);
                if(hex.length()==1) hexString.append('0');
                hexString.append(hex);
            }
        } catch (NoSuchAlgorithmException ex) {
            Logger.getLogger(usuaris.class.getName()).log(Level.SEVERE, null, ex);
        }
        usuari = hexString.toString();

        StringBuffer hexString2 = new StringBuffer();
        try {
            MessageDigest md2 = MessageDigest.getInstance("SHA-256");
            md2.update(autoritzant.getBytes());
            byte byteData[] = md2.digest();
            for (int i=0; i<byteData.length; i++) {
                String hex2=Integer.toHexString(0xff & byteData[i]);
                if(hex2.length()==1) hexString2.append('0');
                hexString2.append(hex2);
            }
        } catch (NoSuchAlgorithmException ex) {
            Logger.getLogger(usuaris.class.getName()).log(Level.SEVERE, null, ex);
        }
        autoritzant = hexString2.toString();

        // Inicialitzo dades de la BD

```

```

String connexioBD = "jdbc:mysql://127.0.0.1/blocjeroni";
Connection connexio;
String con;
ResultSet rs;
String laresposta = "no";
int resultat = 0; // Codi de resposta

// Miro si l'autoritzant Ãs correcte
try {
    Class.forName("com.mysql.jdbc.Driver");
    connexio = (Connection) DriverManager.getConnection(connexioBD, "jeroni","jeroni");
    Statement s = (Statement) connexio.createStatement();
    con = "SELECT admin FROM blocjeroni.usuari WHERE clau=" + autoritzant + ";";
    //System.out.println(con);
    rs = s.executeQuery(con); {
        while (rs.next()) {
            laresposta = rs.getString("admin");
        }
    }
} catch (ClassNotFoundException | SQLException ex) {
    Logger.getLogger(usuaris.class.getName()).log(Level.SEVERE, null, ex);
}

// Aplico canvis d'usuari
if ("si".equals(laresposta)) {
    if ("afegir".equals(estat)) {
        try {
            Class.forName("com.mysql.jdbc.Driver");
            connexio = (Connection) DriverManager.getConnection(connexioBD, "jeroni","jeroni");
            Statement s = (Statement) connexio.createStatement();
            con = "INSERT INTO blocjeroni.usuari (clau, nomaula, admin, descripc) VALUES ('" + usuari + "', '" + aula
+ "', '" + esadmin + "', '" + descripc + "')";
            System.out.println(con);
            s.executeUpdate(con);
            resultat = 1;
        } catch (ClassNotFoundException | SQLException ex) {
            Logger.getLogger(usuaris.class.getName()).log(Level.SEVERE, null, ex);
            resultat = 2;
        }
    } else {
        try {
            Class.forName("com.mysql.jdbc.Driver");
            connexio = (Connection) DriverManager.getConnection(connexioBD, "jeroni","jeroni");
            Statement s = (Statement) connexio.createStatement();
            con = "DELETE FROM blocjeroni.usuari WHERE clau = '" + usuari + "'";
            resultat = 3;
            //System.out.println(con);
            s.executeUpdate(con);
        } catch (ClassNotFoundException | SQLException ex) {
            Logger.getLogger(usuaris.class.getName()).log(Level.SEVERE, null, ex);
            resultat = 1;
        }
    }
}

// Carrego aules
List<String> resposta = new ArrayList<String>();
try {
    Class.forName("com.mysql.jdbc.Driver");
    connexio = (Connection) DriverManager.getConnection(connexioBD, "jeroni","jeroni");
    Statement s = (Statement) connexio.createStatement();
    con = "SELECT nomaula FROM blocjeroni.aula";
    //System.out.println(con);
    rs = s.executeQuery(con); {
        while (rs.next()) {
            resposta.add(rs.getString("nomaula"));
        }
    }
} catch (ClassNotFoundException | SQLException ex) {
    Logger.getLogger(prefiltra.class.getName()).log(Level.SEVERE, null, ex);
}

request.setAttribute("resposta", resposta);
request.setAttribute("resultat", resultat);
getServletContext().getRequestDispatcher("/WEB-INF/bloc/gestiona.jsp").forward(request, response);
}
}

```

Veureusuaris.java

```

package microblog;

import java.io.IOException;
import java.security.MessageDigest;
import java.security.NoSuchAlgorithmException;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;
import java.util.logging.Level;
import java.util.logging.Logger;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import org.json.JSONException;
import org.json.JSONObject;

```

```

/**
 *
 * @author Carles Torras Font
 */
public class veureusuari extends HttpServlet {
    private static final long serialVersionUID = 1L;

    /**
     * Processes requests for both HTTP <code>GET</code> and <code>POST</code>
     * methods.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    protected void processRequest(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        response.setContentType("text/html;charset=UTF-8");
    }

    /**
     * Handles the HTTP <code>POST</code> method.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    @Override
    protected void doPost(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {

        // Pillo dades del JSP
        String autoritzant = request.getParameter("laclau");
        //System.out.println("au: "+autoritzant);
        StringBuffer hexString2 = new StringBuffer();
        try {
            MessageDigest md2 = MessageDigest.getInstance("SHA-256");
            md2.update(autoritzant.getBytes());
            byte byteData[] = md2.digest();
            for (int iiii=0;iiii<byteData.length;iiii++) {
                String hex2=Integer.toHexString(0xff & byteData[iiii]);
                if(hex2.length()==1) hexString2.append('0');
                hexString2.append(hex2);
            }
        } catch (NoSuchAlgorithmException ex) {
            Logger.getLogger(usuari.class.getName()).log(Level.SEVERE, null, ex);
        }
        autoritzant = hexString2.toString();

        // Inicialitzo dades de la BD
        String connexioBD = "jdbc:mysql://127.0.0.1/blocjeroni";
        Connection connexio;
        String con;
        ResultSet rs;
        String resposta = "no";
        JSONObject temporal = new JSONObject();

        // Miro si l'autoritzant Ã©s correcte
        try {
            Class.forName("com.mysql.jdbc.Driver");
            connexio = (Connection) DriverManager.getConnection(connexioBD, "jeroni","jeroni");
            Statement s = (Statement) connexio.createStatement();
            con = "SELECT admin FROM blocjeroni.usuari WHERE clau=" + autoritzant + ";";
            //System.out.println(con);
            rs = s.executeQuery(con); {
                while (rs.next()) {
                    resposta = rs.getString("admin");
                }
            }
        } catch (ClassNotFoundException | SQLException ex) {
            Logger.getLogger(usuari.class.getName()).log(Level.SEVERE, null, ex);
        }

        // Llisto
        if ("si".equals(resposta)) {
            try {
                Class.forName("com.mysql.jdbc.Driver");
                connexio = (Connection) DriverManager.getConnection(connexioBD, "jeroni","jeroni");
                Statement s = (Statement) connexio.createStatement();
                con = "SELECT CONCAT (admin,' ',nomaula,' ',IFNULL(descrip,' ')) AS usuari FROM blocjeroni.usuari;";
                //System.out.println(con);
                rs = s.executeQuery(con); {
                    while (rs.next()) {
                        temporal.append("usuari",rs.getString("usuari"));
                    }
                }
            } catch (ClassNotFoundException | SQLException | JSONException ex) {
                Logger.getLogger(veureusuari.class.getName()).log(Level.SEVERE, null, ex);
            }
            //System.out.println("sortida: "+temporal);
            response.setContentType("application/json");
            response.getWriter().write(temporal.toString());
            processRequest(request, response);
        } else {
            processRequest(request, response);
        }
    }
}

```

```
}  
}  
}
```

A4.2.- Sistema de Gestió de la Informació

Pàgina principal – idex.jsp

Arxiu d'estils CSS – estils.css

Pàgines JSP – Carpeta “bloc”

- Classes.jsp
- Llistaa.jsp
- Llistaassist.jsp
- Llistaau.jsp
- Llistaf.jsp
- Llistainci.jsp
- Llistainf.jsp
- Llistan.jsp
- Llistap.jsp
- Listareu.jsp
- ModificarFotos.jsp
- ModificarInf.jsp
- ModificarReu.jsp
- Modificara.jsp
- Modificarau.jsp
- Modificarn.jsp
- Modificarp.jsp
- NomsAules.jsp
- Index.jsp
- Preindex.jsp

Beans

- Acces.java
- Accesals.java
- Accio.java
- Alumclasse.java
- Alumne.java
- Assistencia.java
- Aula.java
- Categoria.java
- Classes.java
- Cursos.java
- Fotoalumne.java
- Fotografia.java
- Incialumne.java
- Informe.java
- Municipi.java
- Normes.java
- Profclasse.java
- Professional.java
- Profinformes.java
- Reunio.java

JDBC

- Connexio.java

Interface.java
 InterfaceRegistre.java
 OpRegistre.java
 OperacionsBD.java

Paquets font (servlets)

Canvifotografia.java
 Descarrega.java
 GestorAlumnes.java
 GestorAssistència.java
 GestorAules.java
 GestorAules1.java
 GestorClau.java
 GestorFotos.java
 GestorFotos1.java
 GestorIncidencia.java
 GestorInformes.java
 GestorNormes.java
 GestorProfessionals.java
 GestorReunions.java
 Index.java
 TancarSessio.java
 TutIAi.java

Pàgina principal – index.jsp

```
<%--
  Document   : index
  Author    : Dr. Carles Torras Font
--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>Sistema de Gestió del CEE Jeroni de Moragas</title>
    <link rel="stylesheet" href="css/estils.css">
  </head>
  <!-- Script per explicar com han acabat les operacions -->
  <script type="text/javascript">
    window.onload = function() {
      try {
        var resultat = ${resultat};
        if (resultat == 1) {
          alert("Nom d'usuari i/o contrasenya erroris");
        } else if (resultat == 2) {
          alert("L'usuari no és actiu");
        } else if (resultat == 3) {
          alert("No s'ha pogut comprovar si l'usuari és vàlid");
        }
      } catch (error) {}
    };
  </script>
  </head>
  <body>
    <div class="capcalera">
      <table border="0" cellspacing="5" cellpadding="3" align="left">
        <tr><td rowspan="3" align="left"><a href="index.jsp"></a></td><td
        valign="bottom"><b>Benvingut al Sistema de Gestió</b></td></tr>
        <tr><td valign="top"><b>del CEE Jeroni de Moragas!</b></td></tr>
      </table>
    </div>
    <div class="imatge">&nbsp;</div>
    <div class="cos">
      <form action="./Index" method="post">
        NIF: <input type="text" class="form-control" id="usuari" placeholder="nif" name="usuari" size="10"> &nbsp;<input
        type="password" class="form-control" id="clau" placeholder="clau" name="clau"
        size="10"> &nbsp;<input
        type="submit" name="submit" id="submit" value="Publicar" class="boto meu">Entrar</form>
    </div>
```



```

border-radius: 2px 2px 2px 2px;
color: #FFFFFF;
cursor: pointer;
display: inline-block;
font-family: Arial,sans-serif;
font-size: 12px;
font-weight: bold;
line-height: 20px;
margin-bottom: 0;
margin-top: 10px;
padding: 7px 10px;
text-transform: none;
transition: all 0.3s ease 0s;
-moz-transition: all 0.3s ease 0s;
-webkit-transition: all 0.3s ease 0s;
width: 16.795%;
text-align: center;
}
.boto.meu {
background: none repeat scroll 0 0 #999999;
color: #FFFFFF;
}
.boto.meu:hover {
background: none repeat scroll 0 0 greenyellow;
color: gray;
}

.boto2 {
border: 0 none;
border-radius: 2px 2px 2px 2px;
color: #FFFFFF;
cursor: pointer;
display: inline-block;
font-family: Arial,sans-serif;
font-size: 10px;
font-weight: bold;
margin-bottom: 0;
margin-top: 0px;
padding: 3px 3px;
text-transform: none;
transition: all 0.3s ease 0s;
-moz-transition: all 0.3s ease 0s;
-webkit-transition: all 0.3s ease 0s;
text-align: center;
height: 20px;
}
.boto2.meu {
background: none repeat scroll 0 0 #CCCCCC;
color: #FFFFFF;
}
.boto2.meu:hover {
background: none repeat scroll 0 0 greenyellow;
color: #CCCCCC;
}

.btn-sharp {
border-radius:0;
border: 0;
background: none;
-webkit-border-radius: 0px;
-moz-border-radius: 0px;
font-family: 'Comfortaa';
font-size: 100%;
color: #757572;
}

.remove_button_css {
outline: none;
font-family: 'Comfortaa';
line-height: 1.2;
border: 0px;
background-color: transparent;
font-size: 100%;
color: grey;
}
.remove_button_css:hover {
text-decoration: none;
color: black;
}
/*=====*/

#container img {
height: 200px;
width: 200px;
}

#container img {
width: 70%;
height: 70%;
}

/*Bloc d'estils per l'script d'amagar i mostrar blocs de text
=====*/
#wr {
width:800px;
margin: 0px auto;
text-align: left;
}

```

```

.trigger,.trigger a {
    display: block;
    width: 304px;
    height: 26px;
}
.trigger {
    /*background: url(/Alga/css/btn.png) no-repeat 0px 0px;*/
}
.close {
    /*background: url(/Alga/css/btn.png) no-repeat 0px -26px;*/
}
.cnt {
    display: none;
    padding: 10px;
    margin: 10px;
    background: #f9f9f9;
}
.cl {
    width: 100%;
    clear: both;
    line-height: 0px;
    font-size: 0px;
}

/*Bloc d'estils pel login
=====*/
#mask {
    display: none;
    background: white;
    position: fixed; left: 0; top: 0;
    z-index: 10;
    width: 100%; height: 100%;
    opacity: 0.8;
    z-index: 999;
}

.login-popup{
    display:none;
    background: #333;
    padding: 10px;
    border: 2px solid #ddd;
    float: left;
    font-size: 1.2em;
    position: fixed;
    top: 50%; left: 50%;
    z-index: 99999;
    box-shadow: 0px 0px 20px #999;
    -moz-box-shadow: 0px 0px 20px #999; /* Firefox */
    -webkit-box-shadow: 0px 0px 20px #999; /* Safari, Chrome */
    border-radius:3px 3px 3px 3px;
    -moz-border-radius: 3px; /* Firefox */
    -webkit-border-radius: 3px; /* Safari, Chrome */
}

.login-popup2{
    display:none;
    background: white;
    padding: 10px;
    border: 2px solid #ddd;
    float: left;
    font-size: 1.2em;
    position: fixed;
    top: 50%; left: 50%;
    z-index: 99999;
    box-shadow: 0px 0px 20px #999;
    -moz-box-shadow: 0px 0px 20px #999; /* Firefox */
    -webkit-box-shadow: 0px 0px 20px #999; /* Safari, Chrome */
    border-radius:3px 3px 3px 3px;
    -moz-border-radius: 3px; /* Firefox */
    -webkit-border-radius: 3px; /* Safari, Chrome */
}

img.btn_close {float: right; margin: -28px -28px 0 0;}
fieldset { border:none; }
form.signin .textbox label { display:block; padding-bottom:7px; }
form.signin .textbox span {display:block;}
form.signin p, form.signin span {color:#999; font-size:11px; line-height:18px;}
form.signin .textbox input {
    background:#666666;
    border-bottom:1px solid #333;
    border-left:1px solid #000;
    border-right:1px solid #333;
    border-top:1px solid #000;
    color:#fff;
    border-radius: 3px 3px 3px 3px;
    -moz-border-radius: 3px;
    -webkit-border-radius: 3px;
    font:13px Arial, Helvetica, sans-serif;
    padding:6px 6px 4px;
    width:200px;
}

form.signin input:-moz-placeholder { color:#bbb; text-shadow:0 0 2px #000; }
form.signin input::-webkit-input-placeholder { color:#bbb; text-shadow:0 0 2px #000; }

.button {
    background: -moz-linear-gradient(center top, #f3f3f3, #dddddd);
}

```

```

background: -webkit-gradient(linear, left top, left bottom, from(#f3f3f3), to(#dddddd));
background: -o-linear-gradient(top, #f3f3f3, #dddddd);
filter: progid:DXImageTransform.Microsoft.gradient(startColorStr='#f3f3f3', EndColorStr='#dddddd');
border-color:#000;
border-width:1px;
border-radius:4px 4px 4px 4px;
-moz-border-radius: 4px;
-webkit-border-radius: 4px;
color:#333;
cursor:pointer;
display:inline-block;
padding:6px 6px 4px;
margin-top:10px;
font:12px;
width:214px;
}
.button:hover { background:#ddd; }

```

Classes.jsp

```

<!--
Document   : Classes
Created on : 20-mar-2015, 18:31:21
Author    : Dr. Carles Torras Font
-->

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/functions" prefix="fn" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>Gestor d'informació del CEE Jeroni de Moragas</title>

<!--<link href="css/bootstrap.min.css" rel="stylesheet">-->
<link rel="stylesheet" href="css/estils.css">
<script type="text/javascript" src="js/00jquery-1.11.1.js"></script>
<script type="text/javascript" src="js/00amagar.js"></script>

<!-- Script per passar elements d'un costat a l'altre -->
<script>
function swapElement(fromList,toList){
    var selectOptions = document.getElementById(fromList);
    for (var i = 0; i < selectOptions.length; i++) {
        var opt = selectOptions[i];
        if (opt.selected) {
            document.getElementById(fromList).removeChild(opt);
            document.getElementById(toList).appendChild(opt);
            i--;
        }
    }
}
</script>

<!-- Per defecte, el navegador només tornar els elements seleccionats -->
<script type="text/javascript">
function selectAll() {
    selectBox = document.getElementById("right");
    for (var i = 0; i < selectBox.options.length; i++) {
        selectBox.options[i].selected = true;
    }
    selectBox = document.getElementById("right2");
    for (var i = 0; i < selectBox.options.length; i++) {
        selectBox.options[i].selected = true;
    }
}
</script>
</head>
<body>
<div class="capcalera">
<table border="0" cellspacing="5" cellpadding="3" align="left">
<tr><td rowspan="3" align="left"><a href="index.jsp"></a></td><td
valign="bottom"><b>Benvingut al gestor d'informació</b></td></tr>
<tr><td valign="top"><b>del Jeroni de Moragas!</b></td></tr>
</table>
</div>
<div class="imatge">&nbsp;</div>

<div class="cos">
<b>Gestió d'alumnes i tutors de la classe ${aula_nom}, ${aula_curs}</b><br><br>
<div class="avisos">
<form action="./GestorAules1" method="post" class="form-horizontal" accept-charset="utf-8" id="lomeuform"
name="g_classes">
<input type="hidden" name="accio" id="accio" value="classes">
<input type="hidden" name="nomaula" id="nomaula" value="${aula_nom}">
<input type="hidden" name="curs" id="curs" value="${aula_curs}">
<table border="0">
<tr>
<td
<td width='300px'>
Llista d'alumnes
</td>
<td width='20px'></td>
<td width='300px'>

```

```

                Llista d'alumnes de la classe
            </td>
        </tr>
        <tr>
            <td width='300px'>
                <select id="left" multiple="multiple" style="width: 300px;" size="{mida}">
                    <c:forEach items="{llistaalumnes}" var="llistaalumnes">
                        <option value="{llistaalumnes}">{llistaalumnes}</option>
                    </c:forEach>
                </select>
            </td>
            <td width='20px'>
                <input name="&lt;" onclick="swapElement('right','left')" type="button" value="&lt;" />
                <input name="&gt;" onclick="swapElement('left','right')" type="button" value="&gt;" />
            </td>
            <td width='300px'>
                <select id="right" multiple="multiple" name="alclasse" style="width: 300px;" size="{mida}">
                    <c:forEach items="{alumnesaula}" var="alumnesaula">
                        <option value="{alumnesaula}">{alumnesaula}</option>
                    </c:forEach>
                </select>
            </td>
        </tr>
        <tr>
            <td width='300px'>
                <br><br>Llista de tutors
            </td>
            <td width='20px'></td>
            <td width='300px'>
                <br><br>Llista de tutors de la classe
            </td>
        </tr>
        <tr>
            <td width='300px'>
                <select id="left2" multiple="multiple" style="width: 300px;" size="{mida2}">
                    <c:forEach items="{llistatutors}" var="llistatutors">
                        <option value="{llistatutors}">{llistatutors}</option>
                    </c:forEach>
                </select>
            </td>
            <td width='20px'>
                <input name="&lt;" onclick="swapElement('right2','left2')" type="button" value="&lt;" />
                <input name="&gt;" onclick="swapElement('left2','right2')" type="button" value="&gt;" />
            </td>
            <td width='300px'>
                <select id="right2" multiple="multiple" name="tutclasse" style="width: 300px;"
size="{mida2}">
                    <c:forEach items="{profesaula}" var="profesaula">
                        <option value="{profesaula}">{profesaula}</option>
                    </c:forEach>
                </select>
            </td>
        </tr>
    </table>
    <button type="submit" name="submit" id="submit" value="Publicar" class="boto meu"
onclick="selectAll();">Modifica</button>
</form>
</div>
</div>
</body>
</html>

```

Llistaa.jsp

```

<!--
Document   : index
Created on : 20-mar-2015, 18:31:21
Author    : Dr. Carles Torras Font
-->

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/functions" prefix="fn" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>Gestor d'informació del CEE Jeroni de Moragas</title>

<link rel="stylesheet" href="css/estils.css">
<script type="text/javascript" src="js/00jquery-1.11.1.js"></script>
<script type="text/javascript" src="js/00amagar.js"></script>

</head>
<body>
<div class="capcalera">
<table border="0" cellspacing="5" cellpadding="3" align="left">
<tr><td rowspan="3" align="left"><a href="index.jsp"></a></td><td
valign="bottom"><b>Benvingut al gestor d'informació</b></td></tr>
<tr><td valign="top"><b>del Jeroni de Moragas!</b></td></tr>
</table>
</div>
<div class="imatge">&nbsp;</div>

```

```

<div class="cos">
  <b>Llistat d'alumnes</b><br><br>

  <div class="avisos">
    <table border="0" width="10000">
      <thead>
        <tr>
          <th width="50" align="left">&nbsp;</th>
          <th width="70" align="left">&nbsp;</th>
          <th width="200" align="left">Nom</th>
          <th width="200" align="left">Cognom1</th>
          <th width="200" align="left">Cognom2</th>
          <th width="100" align="left">NIF</th>
          <th width="160" align="left">Any entrada</th>
          <th width="120" align="left">Naixement</th>
          <th width="200" align="left">Adreça</th>
          <th width="100" align="left">Població</th>
          <th width="150" align="left">TelÀ fon 1</th>
          <th width="150" align="left">TelÀ fon 2</th>
          <th width="150" align="left">TelÀ fon 3</th>
          <th width="150" align="left">TelÀ fon 4</th>
          <th width="120" align="left">Nom pare</th>
          <th width="120" align="left">Nom mare</th>
          <th width="200" align="left">Email</th>
          <th width="40" align="left">Compartit</th>
          <th width="1300" align="left">DiagnÀstic 1</th>
          <th width="1300" align="left">DiagnÀstic 2</th>
          <th width="1300" align="left">DiagnÀstic 3</th>
          <th width="1300" align="left">DiagnÀstic 4</th>
          <th width="1300" align="left">DiagnÀstic 5</th>
          <th width="1300" align="left">MedicaciÀ 1</th>
          <th width="1300" align="left">MedicaciÀ 2</th>
          <th width="1300" align="left">MedicaciÀ 3</th>
          <th width="1300" align="left">MedicaciÀ 4</th>
          <th width="1300" align="left">MedicaciÀ 5</th>
          <th width="1300" align="left">Pla Individualitzat</th>
        </tr>
      </thead>
      <tbody>
        <c:forEach var="llistaa" items="{llistaa}" varStatus="loop">
          <tr>
            <td><form action="./GestorAlumnes" ENCTYPE="multipart/form-data" class="form-horizontal"
method="post" accept-charset="utf-8" id="lomeuform" name="g_alumnes">
            <input type="hidden" name="nifg" value="{llistaa.nom} {llistaa.cognom1} {llistaa.cognom1},
NIF: {llistaa.nif}">
            <td><button type="submit" name="optionsRadios2" id="submit" value="modificar" class="boto2
meu">Modificar</button></td>
            <td><button type="submit" name="optionsRadios2" id="submit" value="eliminar" class="boto2
meu">Eliminar</button></td>
            </form>
            <td><c:out value="{llistaa.nom}" /></td>
            <td><c:out value="{llistaa.cognom1}" /></td>
            <td><c:out value="{llistaa.cognom2}" /></td>
            <td><c:out value="{llistaa.nif}" /></td>
            <td><c:out value="{llistaa.entrada}" /></td>
            <td><c:out value="{llistaa.naixement}" /></td>
            <td><c:out value="{llistaa.adreca}" /></td>
            <td><c:out value="{llistaa.idmun}" /></td>
            <td><c:out value="{llistaa.telefon1}" /></td>
            <td><c:out value="{llistaa.telefon2}" /></td>
            <td><c:out value="{llistaa.telefon3}" /></td>
            <td><c:out value="{llistaa.telefon4}" /></td>
            <td><c:out value="{llistaa.nompare}" /></td>
            <td><c:out value="{llistaa.nommare}" /></td>
            <td><c:out value="{llistaa.email}" /></td>
            <td><c:out value="{llistaa.compartit}" /></td>
            <td><c:out value="{llistaa.diag1}" /></td>
            <td><c:out value="{llistaa.diag2}" /></td>
            <td><c:out value="{llistaa.diag3}" /></td>
            <td><c:out value="{llistaa.diag4}" /></td>
            <td><c:out value="{llistaa.diag5}" /></td>
            <td><c:out value="{llistaa.med1}" /></td>
            <td><c:out value="{llistaa.med2}" /></td>
            <td><c:out value="{llistaa.med3}" /></td>
            <td><c:out value="{llistaa.med4}" /></td>
            <td><c:out value="{llistaa.med5}" /></td>
            <td><a href="{c:url value="{llistaa.programa}"/></span></a></td>
            <!--Script per eliminar rutes en mostrar els arxius-->
            <script>
              function Abrevio(argument) {
                var n = argument.lastIndexOf("/");
                var res = argument.substring(n+1, argument.length);
                return res;
              }
              document.getElementById("abrevio_{loop.index}").innerHTML = Abrevio('{llistaa.programa}');
            </script>
          </tr>
        </c:forEach>
      </tbody>
    </table> <br>
  </div>
  <form action="./Index" method="post">
    <button type="submit" id="submit" value="Publicar" class="boto meu">Tornar</button>
  </form>
</div>
</body>
</html>

```

Listaassist.jsp

```

<!--
Document   : index
Created on : 20-mar-2015, 18:31:21
Author    : Dr. Carles Torras Font
-->

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/functions" prefix="fn" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>Gestor d'informació del CEE Jeroni de Moragas</title>

<!--<link href="css/bootstrap.min.css" rel="stylesheet"-->
<link rel="stylesheet" href="css/estils.css">
<script type="text/javascript" src="js/00jquery-1.11.1.js"></script>

</head>
<body>
<div class="capcalera">
<table border="0" cellspacing="5" cellpadding="3" align="left">
<tr><td rowspan="3" align="left"><a href="index.jsp"></a></td><td
valign="bottom"><b>Benvingut al gestor d'informació</b></td></tr>
<tr><td valign="top"><b>del Jeroni de Moragas!</b></td></tr>
</table>
</div>
<div class="imatge">&nbsp;</div>

<div class="cos">
<b>Llistat de faltes d'assistència</b><br><br>

<div class="avisos">
<table border="0">
<thead>
<tr>
<th width='70' align='left'>&nbsp;</th>
<th width='80' align='left'>Id</th>
<th width='100' align='left'>Data</th>
<th width='500' align='left'>Motiu</th>
<th width='300' align='left'>Alumne</th>
<th width='300' align='left'>Professional</th>
</tr>
</thead>
<tbody>
<c:forEach var="llistaa" items="${llistaa}" varStatus="status">
<tr>
<td align="right" colspan="6"><form action="./GestorAssistencia" ENCTYPE="multipart/form-data" class="form-horizontal"
method="post" accept-charset="utf-8" id="lomeuform" name="g_assistencia">
<input type="hidden" name="id" value="${llistaa.idassistencia}">

<!-- Inici de la programació dels botons-->
<c:set var="variable" value="false"/>
<c:forEach var="llistaa2" items="{alumnosprofe}" varStatus="status2">
<c:if test="${llistaa2 == fn:substringAfter(nomsalumnos[status.index], ', ')}">
<c:set var="variable" value="true"/>
</c:if>
</c:forEach>
<td>
<button type="submit" name="optionsRadios2" id="submit" value="eliminar"
<c:choose>
<c:when test="${variable == false}">
class="boto2 meu2" disabled
</c:when>
<c:otherwise>
class="boto2 meu"
</c:otherwise>
</c:choose>
Eliminar
</button>
</td>
<!-- Fi de la programació dels botons-->

</form>
<td><c:out value="${llistaa.idassistencia}" /></td>
<td><c:out value="${llistaa.data}" /></td>
<td><c:out value="${llistaa.motiu}" /></td>
<td><c:out value="{nomsalumnos[status.index]}" /></td>
<td><c:out value="{nomsprofes[status.index]}" /></td>
</tr>
</c:forEach>
</tbody>
</table> <br>
</div>
<form action="./Index" method="post">
<button type="submit" id="submit" value="Publicar" class="boto meu">Tornar</button>
</form>
</div>
</body>
</html>

```

Llistau.jsp

```

<!--
  Document   : index
  Created on : 20-mar-2015, 18:31:21
  Author    : Dr. Carles Torras Font
-->

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/functions" prefix="fn" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<!DOCTYPE html>
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>Gestor d'informació del CEE Jeroni de Moragas</title>

    <!--<link href="css/bootstrap.min.css" rel="stylesheet">-->
    <link rel="stylesheet" href="css/estils.css">
    <script type="text/javascript" src="js/00jquery-1.11.1.js"></script>
    <script type="text/javascript" src="js/00amagar.js"></script>

    <!-- Codi del datapicker -->
    <link rel="stylesheet" href="http://code.jquery.com/ui/1.9.0/themes/base/jquery-ui.css" />
    <script src="http://code.jquery.com/jquery-1.8.2.js"></script>
    <script src="http://code.jquery.com/ui/1.9.0/jquery-ui.js"></script>
    <script>
      $(function() {
        $("#datepicker").datepicker({ dateFormat: "yy-mm-dd" });
      });
    </script>
    <script>
      $(function() {
        $("#datepicker2").datepicker({ dateFormat: "yy-mm-dd" });
      });
    </script>

    <!-- Script del veure/amagar -->
    <script type="text/javascript">
      function toggle_visibility(id) {
        var e = document.getElementById(id);
        if(e.style.display == 'block')
          e.style.display = 'none';
        else
          e.style.display = 'block';
      }
    </script>
  </head>
  <body>
    <div class="capcalera">
      <table border="0" cellspacing="5" cellpadding="3" align="left">
        <tr><td rowspan="3" align="left"><a href="index.jsp"></a></td><td
        valign="bottom"><b>Benvingut al gestor d'informació</b></td></tr>
        <tr><td valign="top"><b>del Jeroni de Moragas!</b></td></tr>
      </table>
    </div>
    <div class="imatge">&nbsp;</div>
    <div class="cos">
      <b>Llistat d'aules</b><br><br>
      Clicar damunt el nom de la classe per veure els seus tutors i alumnes <br><br>
    <div class="avisos">
      <table border="0">
        <thead>
          <tr>
            <th width='50' align='left'>&nbsp;</th>
            <th width='50' align='left'>&nbsp;</th>
            <th width='70' align='left'>&nbsp;</th>
            <th width='300' align='left'>Nom</th>
            <th width='370' align='left'>Horari</th>
          </tr>
        </thead>
        <tbody>
          <c:forEach var="llistau" items="${llistau}" varStatus="loop">
            <tr>
              <td><form action="/GestorAules" ENCTYPE="multipart/form-data" class="form-horizontal" method="post"
              accept-charset="utf-8" id="lomeuform" name="g_aules">
                <input type="hidden" name="aula" value="${llistau.nomaula}, ${llistau.curs}">
                <td><button type="submit" name="optionsRadios2" id="submit" value="modificar" class="boto2
              meu">Modificar</button></td>
                <td><button type="submit" name="optionsRadios2" id="submit" value="assignar" class="boto2
              meu">Assignar</button></td>
                <td><button type="submit" name="optionsRadios2" id="submit" value="eliminar" class="boto2
              meu">Eliminar</button></td>
              </form>
            <td><button type="submit" class="btn-sharp" id="veure_${loop.index}">
              <c:out value="${llistau.nomaula}, ${llistau.curs}" /></button></td>
          </tr>
        </tbody>
      </table>
    <!-- Script AJAX per veure tutors/alumnes de la classe -->
    <script type="text/javascript">
      $(document).ready(function() {
        $('#veure_${loop.index}').click(function() {
          var data = {"nomaula": "${llistau.nomaula}", "curs": "${llistau.curs}";
          data = $(this).serialize() + "&" + $.param(data);

```

```

$.ajax({
  type: "POST",
  dataType: "json",
  url: "TutIA1",
  data: data,
  success: function(response) {
    var mida = response.profesaula.length;
    var resultat = "Tutors";
    for (var j = 0; j < mida; j++) {
      resultat = resultat + '\n' + response.profesaula[j];
    }
    //alert("Tutors de l'aula",resultat1);
    var mida = response.alumnesaula.length;
    var resultat = resultat + '\n' + '\n' + "Alumnes";
    for (var j = 0; j < mida; j++) {
      resultat = resultat + '\n' + response.alumnesaula[j];
    }
    alert(resultat);
  }
});
});
});
</script>

<td><a href=<c:url value="{l1listau.lkhorari}" />><span
id="abrevio_{loop.index}></span></a></td>
<!--Script per eliminar rutes en mostrar els arxius-->
<script>
  function Abrevio(argument) {
    var n = argument.lastIndexOf("/");
    var res = argument.substring(n+1, argument.length);
    return res;
  }
  document.getElementById("abrevio_{loop.index}").innerHTML = Abrevio('{l1listau.lkhorari}');
</script>
</tr>
</c:forEach>
</tbody>
</table> <br>
</div>
<form action="./Index" method="post">
  <button type="submit" id="submit" value="Publicar" class="boto meu">Tornar</button>
</form>
</div>
</body>
</html>

```

Llistaf.jsp

```

<!--
Document : index
Created on : 20-mar-2015, 18:31:21
Author : Dr. Carles Torras Font
-->

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/functions" prefix="fn" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>Gestor d'informació del CEE Jeroni de Moragas</title>

<!--<link href="css/bootstrap.min.css" rel="stylesheet"-->
<link rel="stylesheet" href="css/estils.css">
<style type="text/css">
  .imgHover .hover {
    display: none;
    position: absolute;
    z-index: 2;
  }
</style>
<script type="text/javascript" src="js/00jquery-1.11.1.js"></script>
<script type="text/javascript" src="js/00amagar.js"></script>

<!-- Info damunt la fotografia -->
<script type="text/javascript">
  $(window).load(function(){
    $(".imgHover").hover(function() {
      $(this).children("img").fadeOut(200, 0.15)
        .end().children(".hover").show();
    }, function() {
      $(this).children("img").fadeOut(200, 1)
        .end().children(".hover").hide();
    });
  });
</script>

<!-- Codi del datapicker -->
<link rel="stylesheet" href="http://code.jquery.com/ui/1.9.0/themes/base/jquery-ui.css" />
<script src="http://code.jquery.com/jquery-1.8.2.js"></script>
<script src="http://code.jquery.com/ui/1.9.0/jquery-ui.js"></script>
<!-- Programació del datapicker -->
<script>

```



```

                </a>
            </td>
        </c:forEach>
    </tr>
</table> <br>
</div>
<form action="./Index" method="post">
    <button type="submit" id="submit" value="Publicar" class="boto meu">Tornar</button>
</form>
</div>
</body>
</html>

```

Llistainci.jsp

```

<!--
Document : index
Created on : 20-mar-2015, 18:31:21
Author : Dr. Carles Torras Font
-->

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/functions" prefix="fn" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>Gestor d'informació del CEE Jeroni de Moragas</title>

<!--<link href="css/bootstrap.min.css" rel="stylesheet"-->
<link rel="stylesheet" href="css/estils.css">
<script type="text/javascript" src="js/00jquery-1.11.1.js"></script>

</head>
<body>
<div class="capcalera">
<table border="0" cellspacing="5" cellpadding="3" align="left">
<tr><td rowspan="3" align="left"><a href="index.jsp"></a></td><td
valign="bottom"><b>Benvingut al gestor d'informació</b></td></tr>
<tr><td valign="top"><b>del Jeroni de Moragas!</b></td></tr>
</table>
</div>
<div class="imatge">&nbsp;</div>

<div class="cos">
<b>Llistat d'incidències</b><br><br>

<div class="avisos">
<table border="0">
<thead>
<tr>
<th width='70' align='left'>&nbsp;</th>
<th width='80' align='left'>Id</th>
<th width='350' align='left'>Professional</th>
<th width='500' align='left'>TÀ-tol</th>
<th width='250' align='left'>Data</th>
<th width='200' align='left'>Informe</th>
<th width='1000' align='left'>Alumnes</th>
</tr>
</thead>
<tbody>
<c:forEach var="l1listaa" items="${l1listaa}" varStatus="loop">
<tr>
<td align="right"><form action="./GestorIncidencia" ENCTYPE="multipart/form-data" class="form-horizontal"
method="post" accept-charset="utf-8" id="lomeuform" name="g_incidencia">
<input type="hidden" value="" name="id" value="${l1listaa.idincidencia}">

<!-- Inici de la programació dels botons-->
<c:set var="variable" value="false"/>
<c:forEach var="l1listaa3" items="${l1listaa3}" varStatus="status3">
<c:if test="${l1listaa3.idincidencia == l1listaa.idincidencia}">
<c:forEach var="l1listaa2" items="${alumnosprofe}" varStatus="status2">
<c:if test="${l1listaa2 == l1listaa3.nif}">
<c:set var="variable" value="true"/>
</c:if>
</c:forEach>
</c:if>
</c:forEach>
<td align="right">
<button type="submit" name="optionsRadios2" id="submit" value="eliminar"
<c:choose>
<c:when test="${variable == false}">
class="boto2 meu2" disabled
</c:when>
<c:otherwise>
class="boto2 meu"
</c:otherwise>
</c:choose>
Eliminar
</button>
</td>
<!-- Fi de la programació dels botons-->

</form>
<td><c:out value="${l1listaa.idincidencia}" /></td>
<td><c:out value="${nomsprofes[loop.index]}" /></td>
<td><c:out value="${l1listaa.titol}" /></td>

```

```

        <td><c:out value="\${llistaa.data}" /></td>
        <td><a href="\${c:url value="\${llistaa.lkinforme}"/}"><span
id="abrevio_${loop.index}></span></a></td>
        <!--Script per eliminar rutes en mostrar els arxius-->
        <script>
            function Abrevio(argument) {
                var n = argument.lastIndexOf("/");
                var res = argument.substring(n+1, argument.length);
                return res;
            }
            document.getElementById("abrevio_${loop.index}").innerHTML = Abrevio("\${llistaa.lkinforme}");
        </script>
        <td>
            <c:forEach var="llistaal" items="\${llistaal}" varStatus="status">
                <c:if test="\${llistaal.idincidencia == llistaa.idincidencia}">
                    <c:out value="\${nomsalumnes[status.index]}" /> &nbsp;
                </c:if>
            </c:forEach>
        </td>
    </tr>
</tbody>
</table> <br>
</div>
<form action="./Index" method="post">
    <button type="submit" id="submit" value="Publicar" class="boto meu">Tornar</button>
</form>
</div>
</body>
</html>

```

Llistainf.jsp

```

<!--
Document : index
Created on : 20-mar-2015, 18:31:21
Author : Dr. Carles Torras Font
-->

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/functions" prefix="fn" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<!DOCTYPE html>
<html>
<head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>Gestor d'informació del CEE Jeroni de Moragas</title>

    <!--<link href="css/bootstrap.min.css" rel="stylesheet"-->
    <link rel="stylesheet" href="css/estils.css">
    <script type="text/javascript" src="js/00jquery-1.11.1.js"></script>

</head>
<body>
    <div class="capcalera">
        <table border="0" cellspacing="5" cellpadding="3" align="left">
            <tr><td rowspan="3" align="left"><a href="index.jsp"></a></td><td
valign="bottom"><b>Benvingut al gestor d'informació</b></td></tr>
            <tr><td valign="top"><b>del Jeroni de Moragas!</b></td></tr>
        </table>
    </div>
    <div class="imatge">&nbsp;</div>
    <div class="cos">
        <b>Llistat d'informes</b><br><br>
    </div>
    <div class="avisos">
        <table border="0">
            <thead>
                <tr>
                    <th width='50' align='left'>&nbsp;</th>
                    <th width='70' align='left'>&nbsp;</th>
                    <th width='80' align='left'>Id</th>
                    <th width='300' align='left'>Alumne</th>
                    <th width='500' align='left'>TÀ-tol</th>
                    <th width='200' align='left'>Data</th>
                    <th width='300' align='left'>Informe</th>
                    <th width='1500' align='left'>Professionals</th>
                </tr>
            </thead>
            <tbody>
                <c:forEach var="llistaal" items="\${llistaal}" varStatus="loop">
                    <tr>
                        <form action="./GestorInformes" ENCTYPE="multipart/form-data" class="form-horizontal"
method="post" accept-charset="utf-8" id="lomeuform" name="g_informes">
                            <input type="hidden" name="id" value="\${llistaal.idinforme}">
                            <!-- Inici de la programació dels botons-->
                            <c:set var="variable" value="false"/>
                            <c:forEach var="llistaal2" items="\${alumnesprofe}" varStatus="status2">
                                <c:if test="\${llistaal2 == fn:substringAfter(nomsalumnes[loop.index], ', ')}">
                                    <c:set var="variable" value="true"/>
                                </c:if>
                            </c:forEach>
                        <td>
                            <button type="submit" name="optionsRadios2" id="submit" value="modificar"
                                <c:choose>

```

```

        <c:when test="${variable == false}">
            class="boto2 meu2" disabled
        </c:when>
        <c:otherwise>
            class="boto2 meu"
        </c:otherwise>
    </c:choose>
    Modificar
</button>
</td>
<td>
    <button type="submit" name="optionsRadios2" id="submit" value="eliminar"
    <c:choose>
        <c:when test="${variable == false}">
            class="boto2 meu2" disabled
        </c:when>
        <c:otherwise>
            class="boto2 meu"
        </c:otherwise>
    </c:choose>
    Eliminar
</button>
</td>
<!-- Fi de la programació dels botons-->
</form>
<td><c:out value="${llistaa.idinforme}" /></td>
<td><c:out value="${nomsalumes[loop.index]}" /></td>
<td><c:out value="${llistaa.titolinforme}" /></td>
<td><c:out value="${llistaa.data}" /></td>
<td><a href="<c:url value="${llistaa.lkinforme}"/>"><span
id="abrevio_${loop.index}"></span></a></td>
<!--Script per eliminar rutes en mostrar els arxius-->
<script>
    function Abrevio(argument) {
        var n = argument.lastIndexOf("/");
        var res = argument.substring(n+1, argument.length);
        return res;
    }
    document.getElementById("abrevio_${loop.index}").innerHTML = Abrevio('${llistaa.lkinforme}');
</script>
<td>
    <c:forEach var="llistaal" items="${llistaal}" varStatus="status">
        <c:if test="${llistaal.idinforme == llistaa.idinforme}">
            <c:out value="${nomsprofes[status.index]}" /> &nbsp;
        </c:if>
    </c:forEach>
</td>
</tr>
</c:forEach>
</tbody>
</table> <br>
</div>
<form action="./Index" method="post">
    <button type="submit" id="submit" value="Publicar" class="boto meu">Tornar</button>
</form>
</div>
</body>
</html>

```

Llistan.jsp

```

<%--
Document : index
Created on : 20-mar-2015, 18:31:21
Author : Dr. Carles Torras Font
--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/functions" prefix="fn" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>Gestor d'informació del CEE Jeroni de Moragas</title>

<!--<link href="css/bootstrap.min.css" rel="stylesheet"-->
<link rel="stylesheet" href="css/estils.css">
<script type="text/javascript" src="js/00jquery-1.11.1.js"></script>
<script type="text/javascript" src="js/00amagar.js"></script>

<!-- Codi del datapicker -->
<link rel="stylesheet" href="http://code.jquery.com/ui/1.9.0/themes/base/jquery-ui.css" />
<script src="http://code.jquery.com/jquery-1.8.2.js"></script>
<script src="http://code.jquery.com/ui/1.9.0/jquery-ui.js"></script>
<script>
    $(function() {
        $("#datepicker" ).datepicker({ dateFormat: "yy-mm-dd" });
    });
</script>
<script>
    $(function() {
        $("#datepicker2" ).datepicker({ dateFormat: "yy-mm-dd" });
    });
</script>

```

```

<!-- Script del veure/amagar -->
<script type="text/javascript">
    function toggle_visibility(id) {
        var e = document.getElementById(id);
        if(e.style.display == 'block')
            e.style.display = 'none';
        else
            e.style.display = 'block';
    }
</script>
</head>
<body>
    <div class="capcalera">
        <table border="0" cellspacing="5" cellpadding="3" align="left">
            <tr><td rowspan="3" align="left"><a href="index.jsp"></a></td><td
            valign="bottom"><b>Benvingut al gestor d'informació</b></td></tr>
            <tr><td valign="top"><b>del Jeroni de Moragas!</b></td></tr>
        </table>
    </div>
    <div class="imatge">&nbsp;</div>

    <div class="cos">
        <b>Llistat de normes</b><br><br>

        <div class="avisos">
            <table border="0">
                <thead>
                    <tr>
                        <th width='50' align='left'>&nbsp;</th>
                        <th width='70' align='left'>&nbsp;</th>
                        <th width='80' align='left'>Id</th>
                        <th width='200' align='left'>TÀ-tol</th>
                        <th width='370' align='left'>Arxiu</th>
                    </tr>
                </thead>
                <tbody>
                    <c:forEach var="llistan" items="{l1listan}" varStatus="loop">
                        <tr>
                            <td><form action="./GestorNormes" ENCTYPE="multipart/form-data" class="form-horizontal" method="post"
                            accept-charset="utf-8" id="lomeuform" name="g_normes">
                                <input type="hidden" name="id" value="{l1listan.idnormes}">
                                <td><button type="submit" name="optionsRadios2" id="submit" value="modificar" class="boto2
                                meu">Modificar</button></td>
                                <td><button type="submit" name="optionsRadios2" id="submit" value="eliminar" class="boto2
                                meu">Eliminar</button></td>
                            </form>
                            <td><c:out value="{l1listan.idnormes}" /></td>
                            <td><c:out value="{l1listan.titolnorma}" /></td>
                            <td><a href="{c:url value="{l1listan.lknorma}"/>"><span
                            id="abrevio_{loop.index}"></span></a></td>
                            <!--Script per eliminar rutes en mostrar els arxius-->
                            <script>
                                function Abrevio(argument) {
                                    var n = argument.lastIndexOf("/");
                                    var res = argument.substring(n+1, argument.length);
                                    return res;
                                }
                                document.getElementById("abrevio_{loop.index}").innerHTML = Abrevio('{l1listan.lknorma}');
                            </script>
                            </td>
                        </c:forEach>
                    </tbody>
                </table> <br>
            </div>
            <form action="./Index" method="post">
                <button type="submit" id="submit" value="Publicar" class="boto meu">Tornar</button>
            </form>
        </div>
    </body>
</html>

```

Llistap.jsp

```

<!--
Document   : index
Created on : 20-mar-2015, 18:31:21
Author    : Dr. Carles Torras Font
-->

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/functions" prefix="fn" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>Gestor d'informació del CEE Jeroni de Moragas</title>

        <!--<link href="css/bootstrap.min.css" rel="stylesheet"-->
        <link rel="stylesheet" href="css/estils.css">
        <script type="text/javascript" src="js/00jquery-1.11.1.js"></script>
        <script type="text/javascript" src="js/00amagar.js"></script>

    </head>
    <body>
        <div class="capcalera">
            <table border="0" cellspacing="5" cellpadding="3" align="left">

```

```

        <tr><td rowspan="3" align="left"><a href="index.jsp"></a></td><td
valign="bottom"><b>Benvingut al gestor d'informació</b></td></tr>
        <tr><td valign="top"><b>del Jeroni de Moragas!</b></td></tr>
    </table>
</div>
<div class="imatge">&nbsp;</div>

<div class="cos">
    <b>Llistat de professionals</b><br><br>

    <div class="avisos">
        <table border="0">
            <thead>
                <tr>
                    <th width='50' align='left'>&nbsp;</th>
                    <th width='70' align='left'>&nbsp;</th>
                    <th width='100' align='left'>Nom</th>
                    <th width='150' align='left'>Cognom1</th>
                    <th width='150' align='left'>Cognom2</th>
                    <th width='100' align='left'>NIF</th>
                    <th width='50' align='left'>Actiu</th>
                    <th width='100' align='left'>Tutoria</th>
                    <th width='370' align='left'>Horari</th>
                </tr>
            </thead>
            <tbody>
                <c:forEach var="llistap" items="${llistap}" varStatus="loop">
                    <tr>
                        <td><form action="./GestorProfessionals" ENCTYPE="multipart/form-data" class="form-horizontal"
method="post" accept-charset="utf-8" id="lomeuform" name="g_professionals">
                            <input type="hidden" name="nifg" value="${llistap.nom} ${llistap.cognom1} ${llistap.cognom2},
NIF: ${llistap.nif}">
                            <td><button type="submit" name="optionsRadios2" id="submit" value="modificar" class="boto2
meu">Modificar</button></td>
                            <td><button type="submit" name="optionsRadios2" id="submit" value="eliminar" class="boto2
meu">Eliminar</button></td>
                        <td><c:out value="${llistap.nom}" /></td>
                        <td><c:out value="${llistap.cognom1}" /></td>
                        <td><c:out value="${llistap.cognom2}" /></td>
                        <td><c:out value="${llistap.nif}" /></td>
                        <td><c:out value="${llistap.actiu}" /></td>
                        <td><c:out value="${llistap.tutoria}" /></td>
                        <td><a href="c:url value="${llistap.lkhorari}"/></span></a></td>
                    </tr>
                </c:forEach>
            </tbody>
        </table> <br>
    </div>
    <form action="./Index" method="post">
        <button type="submit" id="submit" value="Publicar" class="boto meu">Tornar</button>
    </form>
</div>
</body>
</html>

```

Llistareu.jsp

```

<!--
Document : index
Created on : 20-mar-2015, 18:31:21
Author : Dr. Carles Torras Font
-->

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/functions" prefix="fn" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>Gestor d'informació del CEE Jeroni de Moragas</title>

        <!--<link href="css/bootstrap.min.css" rel="stylesheet"-->
        <link rel="stylesheet" href="css/estils.css">
        <script type="text/javascript" src="js/00jquery-1.11.1.js"></script>

    </head>
    <body>
        <div class="capcalera">
            <table border="0" cellspacing="5" cellpadding="3" align="left">
                <tr><td rowspan="3" align="left"><a href="index.jsp"></a></td><td
valign="bottom"><b>Benvingut al gestor d'informació</b></td></tr>
                <tr><td valign="top"><b>del Jeroni de Moragas!</b></td></tr>
            </table>
        </div>
        <div class="imatge">&nbsp;</div>

```

```

<div class="cos">
  <b>Llistat de reunions</b><br><br>
  <div class="avisos">
    <table border="0">
      <thead>
        <tr>
          <th width='50' align='left'>&nbsp;</th>
          <th width='70' align='left'>&nbsp;</th>
          <th width='80' align='left'>Id</th>
          <th width='300' align='left'>Alumne</th>
          <th width='500' align='left'>Assistents</th>
          <th width='300' align='left'>Data i hora</th>
          <th width='200' align='left'>Informe</th>
          <th width='80' align='left'>Anterior</th>
          <th width='1500' align='left'>Professionals</th>
        </tr>
      </thead>
      <tbody>
        <c:forEach var="l1istaa" items="{l1istaa}" varStatus="loop">
          <tr>
            <td>
              <form action="./GestorReunions" ENCTYPE="multipart/form-data" class="form-horizontal"
method="post" accept-charset="utf-8" id="lomeuform" name="g_reunions">
                <input type="hidden" name="id" value="{l1istaa.idreunio}">
                <!-- Inici de la programació dels botons-->
                <c:set var="variable" value="false"/>
                <c:forEach var="l1istaa2" items="{alumesprofe}" varStatus="status2">
                  <c:if test="{l1istaa2 == fn:substringAfter(nomsalumes[loop.index], ', ')}">
                    <c:set var="variable" value="true"/>
                  </c:if>
                </c:forEach>
                <td>
                  <button type="submit" name="optionsRadios2" id="submit" value="modificar"
                    <c:choose>
                      <c:when test="{variable == false}">
                        class="boto2 meu2" disabled
                      </c:when>
                      <c:otherwise>
                        class="boto2 meu"
                      </c:otherwise>
                    </c:choose>>
                    Modificar
                  </button>
                </td>
                <td>
                  <button type="submit" name="optionsRadios2" id="submit" value="eliminar"
                    <c:choose>
                      <c:when test="{variable == false}">
                        class="boto2 meu2" disabled
                      </c:when>
                      <c:otherwise>
                        class="boto2 meu"
                      </c:otherwise>
                    </c:choose>>
                    Eliminar
                  </button>
                </td>
                <!-- Fi de la programació dels botons-->
                </form>
                <td><c:out value="{l1istaa.idreunio}" /></td>
                <td><c:out value="{nomsalumes[loop.index]}" /></td>
                <td><c:out value="{l1istaa.assistents}" /></td>
                <td><c:out value="{l1istaa.datahora}" /></td>
                <td><a href="{c:url value="{l1istaa.lkcontingut}"/}><span
id="abrevio_{loop.index}"></span></a></td>
                <!--Script per eliminar rutes en mostrar els arxius-->
                <script>
                  function Abrevio(argument) {
                    var n = argument.lastIndexOf("/");
                    var res = argument.substring(n+1, argument.length);
                    return res;
                  }
                document.getElementById("abrevio_{loop.index}").innerHTML =
Abrevio("{l1istaa.lkcontingut}");
                </script>
                <td><c:out value="{l1istaa.propera}" /></td>
                <td>
                  <c:forEach var="l1istaal" items="{l1istaal}" varStatus="status">
                    <c:if test="{l1istaal.idreunio == l1istaa.idreunio}">
                      <c:out value="{nomsprofes[status.index]}" /> &nbsp;&nbsp;&nbsp;
                    </c:if>
                  </c:forEach>
                </td>
              </tr>
            </c:forEach>
          </tbody>
        </table> <br>
      </div>
      <form action="./Index" method="post">
        <button type="submit" id="submit" value="Publicar" class="boto meu">Tornar</button>
      </form>
    </div>
  </body>
</html>

```

ModificarFotos.jsp

```

<!--
  Document   : index
  Created on : 20-mar-2015, 18:31:21
  Author    : Dr. Carles Torras Font
-->

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/functions" prefix="fn" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<!DOCTYPE html>
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>Gestor d'informació del CEE Jeroni de Moragas</title>

    <!--<link href="css/bootstrap.min.css" rel="stylesheet"-->
    <link rel="stylesheet" href="css/estils.css">
    <script type="text/javascript" src="js/00jquery-1.11.1.js"></script>
    <script type="text/javascript" src="js/00amagar.js"></script>

    <!-- Codi del datapicker -->
    <link rel="stylesheet" href="http://code.jquery.com/ui/1.9.0/themes/base/jquery-ui.css" />
    <script src="http://code.jquery.com/jquery-1.8.2.js"></script>
    <script src="http://code.jquery.com/ui/1.9.0/jquery-ui.js"></script>

    <!-- Codi de l'add-on time-picker al datepicker -->
    <link rel="stylesheet" href="css/jquery-ui-timepicker-addon.css">
    <script type="text/javascript" src="js/jquery-ui-timepicker-addon.js"></script>

    <script>
      $(function() {
        $('#datepicker8').datetimepicker({ dateFormat: "yy-mm-dd", timeFormat: "hh:mm:ss" });
      });
    </script>
  </head>
  <body>
    <div class="capcalera">
      <table border="0" cellspacing="5" cellpadding="3" align="left">
        <tr><td rowspan="3" align="left"><a href="index.jsp"></a></td><td
        valign="bottom"><b>Benvingut al gestor d'informació</b></td></tr>
        <tr><td align="top"><b>del Jeroni de Moragas!</b></td></tr>
      </table>
    </div>
    <div class="imatge">&nbsp;</div>
    <div class="cos">
      <b>Modificar fotografies</b><br><br>

      <form action="./GestorFotos" method="post" enctype="multipart/form-data" class="form-horizontal" accept-
      charset="utf-8" id="lomeuform" name="g_fotos">
        <input type="radio" name="optionsRadios2" id="optionsRadios2" value="modificarex" checked hidden>
        <input type="hidden" name="id" id="id" value=${fotografia.id}>
        <div id="veuprof">
          <div class="form-group">
            <div class="form-group">
              Aula
              <select id="Select" class="form-control" name="aula" style="width: 200px">
                <c:forEach items=${aules} var="option">
                  <option value=${option} ${option == lonomdaula ? 'selected' : ''}>${option}</option>
                </c:forEach>
              </select>
            </div><br>
            <div class="col-sm-2">
              Data <input type="text" class="form-control" id="datepicker8" name="data" size="20"
              value=${fotografia.data}>
            </div><br>
            <div class="col-sm-2">
              Etiqueta 1 <input type="text" class="form-control" id="et1" name="et1" size="13"
              value=${fotografia.etiqueta1}>
              Etiqueta 2 <input type="text" class="form-control" id="et2" name="et2" size="13"
              value=${fotografia.etiqueta2}>
              Etiqueta 3 <input type="text" class="form-control" id="et3" name="et3" size="13"
              value=${fotografia.etiqueta3}>
              Etiqueta 4 <input type="text" class="form-control" id="et4" name="et4" size="13"
              value=${fotografia.etiqueta4}>
            </div><br>
            <div class="form-group">
              Alumnes relacionats amb la fotografia<br><br>
              <select id="Select" multiple="multiple" class="form-control" name="alumnes" style="width: 300px">
                <c:forEach items=${elsalumnes} var="option13" varStatus="status">
                  <option value=${option13}>
                    <c:forEach items=${l1listafal} var="option15">
                      ${elsalumnesnif[status.index] == option15 ? 'selected' : ''}
                    </c:forEach>
                  >${option13}</option>
                </c:forEach>
              </select>
            </div><br>
            <div class="col-sm-2">
              <img src=${fotografia.lkfoto}><br><br>
              <input type="hidden" class="form-control" id="fotoactual" name="fotoactual" size="13"
              value=${fotografia.lkfoto}>
              Arxiu fotogrÀfic: <input type="file" name="uploadFile">
            </div><br>

```



```

                </c:forEach>
                >${option14}
            </option>
        </c:forEach>
    </select>
</div><br>
<div class="col-sm-2">
    Assistentes <input type="text" class="form-control" id="assistents" name="assistents" size="100"
value=${reunio.assistents}>
</div><br>
<div class="col-sm-2">
    Data i hora <input type="text" class="form-control" id="datepicker7" name="datahora" size="15"
value=${reunio.datahora}>
</div><br>
Informe de la reunió:
<a href=${reunio.lkcontingut}>
    <span id="abrevio">
    </span>
<!--Script per eliminar rutes en mostrar els arxius-->
<script>
    function Abrevio(argument) {
        var n = argument.lastIndexOf("/");
        var res = argument.substring(n+1, argument.length);
        return res;
    }
    document.getElementById("abrevio").innerHTML = Abrevio('${reunio.lkcontingut}');
</script>
</a>
<input type='hidden' name='lkcontingut' value='${reunio.lkcontingut}'> <br><br>
<div class="col-sm-2">
    Nou informe de la reunió: <input type="file" name="uploadFile">
</div><br>
<div class="col-sm-2">
    Reunió anterior:
    <label>
        <input type="radio" name="optionsRadios3" id="optionsRadios3" value="nonhiha"
${reunio.propera=='0'?checked:''}>&nbsp; No n'hi ha cap.
    </label>
    <label>
        <input type="radio" name="optionsRadios3" id="optionsRadios3" value="sinhiha"
${reunio.propera!='0'?checked:''}>&nbsp; N'hi ha:
    </label>
    <select id="Select" class="form-control" name="idanterior" style="width: 100px">
        <c:forEach items="{reunions}" var="option20">
            <option value="{option20}" ${option20 == reunio.propera ? 'selected' :
''}>${option20}</option>
        </c:forEach>
    </select>
</div>
</div>
<div>
    <button type="submit" name="submit" id="submit" value="Publicar" class="boto meu">Modifica</button>
</div>
</form>
</div>
</body>
</html>

```

Modificara.jsp

```

<!--
    Document    : Modificara
    Created on  : 20-mar-2015, 18:31:21
    Author      : Dr. Carles Torras Font
-->

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/functions" prefix="fn" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<!DOCTYPE html>
<html>
<head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>Gestor d'informació del CEE Jeroni de Moragas</title>

    <!--<link href="css/bootstrap.min.css" rel="stylesheet">-->
    <link rel="stylesheet" href="css/estils.css">
    <script type="text/javascript" src="js/00jquery-1.11.1.js"></script>
    <script type="text/javascript" src="js/00amagar.js"></script>

    <!-- Codi del datapicker -->
    <link rel="stylesheet" href="http://code.jquery.com/ui/1.9.0/themes/base/jquery-ui.css" />
    <script src="http://code.jquery.com/jquery-1.8.2.js"></script>
    <script src="http://code.jquery.com/ui/1.9.0/jquery-ui.js"></script>
    <script>
        $(function() {
            $( "#datepicker" ).datepicker({ dateFormat: "yy-mm-dd" });
        });
    </script>
    <script>
        $(function() {
            $( "#datepicker2" ).datepicker({ dateFormat: "yy-mm-dd" });
        });
    </script>
</head>
<body>

```

```

<div class="capcalera">
  <table border="0" cellspacing="5" cellpadding="3" align="left">
    <tr><td rowspan="3" align="left"><a href="index.jsp"></a></td><td
valign="bottom"><b>Benvingut al gestor d'informació</b></td></tr>
    <tr><td valign="top"><b>del Jeroni de Moragas!</b></td></tr>
  </table>
</div>
<div class="imatge">&nbsp;</div>

<div class="cos">
  <b>Modificar alumne</b><br><br>

  <form action="./GestorAlumnes" method="post" enctype="multipart/form-data" class="form-horizontal" accept-
charset="utf-8" id="lomeuform" name="g_alumnes">
  <input type="radio" name="optionsRadios2" id="optionsRadios2" value="modificarex" checked hidden>
  <div class="form-group">
    <div class="col-sm-2">
      Nom <input type="text" class="form-control" id="data" name="nom" size="15" value="{alumne.nom}">
      1r cognom <input type="text" class="form-control" id="data" name="cognom1" size="15"
value="{alumne.cognom1}">
      2n cognom <input type="text" class="form-control" id="data" name="cognom2" size="15"
value="{alumne.cognom2}">
    </div><br>
    <div class="col-sm-2">
      NIF <input type="text" class="form-control" id="data" name="nifa" size="15" value="{alumne.nif}"
readonly>
      Data de naixement <input type="text" class="form-control" id="datepicker" name="datanaix" size="6"
value="{alumne.naixement}">
      Any d'entrada <input type="text" class="form-control" id="data" name="entrada" size="6"
value="{alumne.entrada}">
    </div><br>
    <div class="col-sm-2">
      Adreça <input type="text" class="form-control" id="adreça" name="adreca" size="35"
value="{alumne.adreca}">
      Població
      <select id="Select" class="form-control" name="poblacio" style="width: 300px">
        <c:forEach items="{municipis}" var="option">
          <option value="{option}"
            >{option == municipi ? 'selected' : ''}
          >{option}</option>
        </c:forEach>
      </select>
    </div><br>
    <div class="col-sm-2">
      Telèfon 1 <input type="text" class="form-control" id="tel1" name="tel1" size="13"
value="{alumne.telefon1}">
      Telèfon 2 <input type="text" class="form-control" id="tel2" name="tel2" size="13"
value="{alumne.telefon2}">
      Telèfon 3 <input type="text" class="form-control" id="tel3" name="tel3" size="13"
value="{alumne.telefon3}">
      Telèfon 4 <input type="text" class="form-control" id="tel4" name="tel4" size="13"
value="{alumne.telefon4}">
    </div><br>
    <div class="col-sm-2">
      Nom del pare <input type="text" class="form-control" id="pare" name="pare" size="13"
value="{alumne.nompare}">
      Nom de la mare <input type="text" class="form-control" id="mare" name="mare" size="13"
value="{alumne.nommare}">
      Nom del tutor/a <input type="text" class="form-control" id="mare" name="nomtutor" size="13"
value="{alumne.nomtutortutora}">
    </div><br>
    <div class="col-sm-2">
      Nom de la parella del pare <input type="text" class="form-control" id="pare" name="parellapare"
size="13" value="{alumne.nomparellapare}">
      Nom de la parella de la mare <input type="text" class="form-control" id="mare" name="parellamare"
size="13" value="{alumne.nomparellamare}">
    </div><br>
    <div class="col-sm-2">
      NÀª de targeta sanitària <input type="text" class="form-control" id="email" name="numtarsan"
size="18" value="{alumne.numtarsan}">
      Email <input type="text" class="form-control" id="email" name="email" size="13"
value="{alumne.email}">
    </div><br>
    <div class="col-sm-2">
      Diagnòstic 1 <input type="text" class="form-control" id="diag1" name="diag1" size="90"
value="{alumne.diag1}">
    </div><br>
    <div class="col-sm-2">
      Diagnòstic 2 <input type="text" class="form-control" id="diag2" name="diag2" size="90"
value="{alumne.diag2}">
    </div><br>
    <div class="col-sm-2">
      Diagnòstic 3 <input type="text" class="form-control" id="diag3" name="diag3" size="90"
value="{alumne.diag3}">
    </div><br>
    <div class="col-sm-2">
      Diagnòstic 4 <input type="text" class="form-control" id="diag4" name="diag4" size="90"
value="{alumne.diag4}">
    </div><br>
    <div class="col-sm-2">
      Diagnòstic 5 <input type="text" class="form-control" id="diag5" name="diag5" size="90"
value="{alumne.diag5}">
    </div><br>
    <div class="col-sm-2">
      Medicació 1 <input type="text" class="form-control" id="med1" name="med1" size="90"
value="{alumne.med1}">
    </div><br>
  </div><br>
  <div class="col-sm-2">

```

```

                Medicació 2 <input type="text" class="form-control" id="med2" name="med2" size="90"
value=${alumne.med2}>
            </div><br>
            <div class="col-sm-2">
                Medicació 3 <input type="text" class="form-control" id="med3" name="med3" size="90"
value=${alumne.med3}>
            </div><br>
            <div class="col-sm-2">
                Medicació 4 <input type="text" class="form-control" id="med4" name="med4" size="90"
value=${alumne.med4}>
            </div><br>
            <div class="col-sm-2">
                Medicació 5 <input type="text" class="form-control" id="med5" name="med5" size="90"
value=${alumne.med5}>
            </div><br>
            Programa d'enllaç:
            <a href=${alumne.programa}>
                <span id="abrevio">
                </span>
            <!--Script per eliminar rutes en mostrar els arxius-->
            <script>
                function Abrevio(argument) {
                    var n = argument.lastIndexOf("/");
                    var res = argument.substring(n+1, argument.length);
                    return res;
                }
                document.getElementById("abrevio").innerHTML = Abrevio('${alumne.programa}');
            </script>
            </a>
            <input type='hidden' name='programa' value='${alumne.programa}'> <br><br>
            <div class="col-sm-2">
                Arxiu del Pla Individualitzat: <input type="file" name="uploadFile">
            </div><br>
            <div class="radio">
                <label>
                    <input type="radio" name="optionsRadios1" id="optionsRadios1" value="si"
${alumne.compartit=='si'? 'checked':''}> És compartit
                </label>
                <label>
                    <input type="radio" name="optionsRadios1" id="optionsRadios2" value="no"
${alumne.compartit=='no'? 'checked':''}> No és compartit
                </label>
            </div><br>
            </div>
            <button type="submit" name="submit" id="submit" value="Publicar" class="boto meu">Modifica</button>
        </form>
    </div>
</body>
</html>

```

Modificarau.jsp

```

<!--
Document : Modificarau
Created on : 20-mar-2015, 18:31:21
Author : Dr. Carles Torras Font
-->

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/functions" prefix="fn" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<!DOCTYPE html>
<html>
<head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>Gestor d'informació del CEE Jeroni de Moragas</title>

    <!--<link href="css/bootstrap.min.css" rel="stylesheet"-->
    <link rel="stylesheet" href="css/estils.css">
    <script type="text/javascript" src="js/00jquery-1.11.1.js"></script>
    <script type="text/javascript" src="js/00amagar.js"></script>

    <!--Script per eliminar rutes en mostrar els arxius-->
    <script>
        function Abrevio() {
            var str = "${aula.lkhorari}";
            var n = str.lastIndexOf("/");
            var res = str.substring(n+1, str.length);
            document.getElementById("abrevio").innerHTML = res;
        }
    </script>

</head>
<body onload="Abrevio();">
    <div class="capcalera">
        <table border="0" cellspacing="5" cellpadding="3" align="left">
            <tr><td rowspan="3" align="left"><a href="index.jsp"></a></td><td
valign="bottom"><b>Benvingut al gestor d'informació</b></td></tr>
            <tr><td valign="top"><b>del Jeroni de Moragas!</b></td></tr>
        </table>
    </div>
    <div class="imatge">&nbsp;</div>
    <div class="cos">
        <b>Modificar classe</b><br><br>

```

```

    <form action="/GestorAules" method="post" enctype="multipart/form-data" class="form-horizontal" accept-
charset="utf-8" id="lomeuform" name="g_aules">
      <input type="radio" name="optionsRadios2" id="optionsRadios2" value="modificarex" checked hidden>
      <div id="veuprof">
        <div class="form-group">
          <div class="col-sm-2">
            Aula <input type="text" class="form-control" id="data" name="aula" size="25" readonly
value="{nomaula}">
            Horari de l'aula: <a href="{aula.lkhorari}"><span id="abrevio"></span></a>
            <input type="hidden" name="horariaula" value="{aula.lkhorari}">
          </div><br>
          <div class="col-sm-2">
            Nou arxiu d'horaris de l'aula: <input type="file" name="uploadFile">
          </div><br>
        </div>
      </div>
      <button type="submit" name="submit" id="submit" value="Publicar" class="boto meu">Modifica</button>
    </form>
  </div>
</body>
</html>

```

Modificarn.jsp

```

<!--
Document : Modificarn
Created on : 20-mar-2015, 18:31:21
Author : Dr. Carles Torras Font
-->

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/functions" prefix="fn" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<!DOCTYPE html>
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>Gestor d'informació del CEE Jeroni de Moragas</title>

    <!--<link href="css/bootstrap.min.css" rel="stylesheet"-->
    <link rel="stylesheet" href="css/estils.css">
    <script type="text/javascript" src="js/00jquery-1.11.1.js"></script>
    <script type="text/javascript" src="js/00amagar.js"></script>

  </head>
  <body>
    <div class="capcalera">
      <table border="0" cellspacing="5" cellpadding="3" align="left">
        <tr><td rowspan="3" align="left"><a href="index.jsp"></a></td><td
valign="bottom"><b>Benvingut al gestor d'informació</b></td></tr>
        <tr><td valign="top"><b>del Jeroni de Moragas!</b></td></tr>
      </table>
    </div>
    <div class="imatge">&nbsp;</div>

    <div class="cos">
      <b>Modificar norma</b><br><br>

      <form action="/GestorNormes" method="post" enctype="multipart/form-data" class="form-horizontal" accept-
charset="utf-8" id="lomeuform" name="g_normes">
        <input type="radio" name="optionsRadios2" id="optionsRadios2" value="modificarex" checked hidden>
        <div id="veuprof">

          <div class="form-group">
            <div class="col-sm-2">
              <input type="hidden" class="form-control" id="id" name="id" size="75" value="{norma.idnormes}">
            </div>
            <div class="col-sm-2">
              TÀ-tol <input type="text" class="form-control" id="titol" name="titol" size="75"
value="{norma.titolnorma}">
            </div><br>
            Arxiu explicatiu de la norma:
            <a href="{norma.lknorma}">
              <span id="abrevio">
            </span>
            <!--Script per eliminar rutes en mostrar els arxius-->
            <script>
              function Abrevio(argument) {
                var n = argument.lastIndexOf("/");
                var res = argument.substring(n+1, argument.length);
                return res;
              }
              document.getElementById("abrevio").innerHTML = Abrevio("{norma.lknorma}");
            </script>
            </a>
            <input type="hidden" name="lknorma" value="{norma.lknorma}"> <br><br>
            <div class="col-sm-2">
              Nou arxiu explicatiu de la norma: <input type="file" name="uploadFile">
            </div><br>
          </div>
        </div>
        <button type="submit" name="submit" id="submit" value="Publicar" class="boto meu">Modifica</button>
      </form>
    </div>

```

```

</body>
</html>

```

Modificarp.jsp

```

<%--
Document : Modificarp
Created on : 20-mar-2015, 18:31:21
Author : Dr. Carles Torras Font
--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/functions" prefix="fn" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>Gestor d'informació del CEE Jeroni de Moragas</title>

<!--<link href="css/bootstrap.min.css" rel="stylesheet">-->
<link rel="stylesheet" href="css/estils.css">
<script type="text/javascript" src="js/00jquery-1.11.1.js"></script>
<script type="text/javascript" src="js/00amagar.js"></script>

<!-- Codi del datapicker -->
<link rel="stylesheet" href="http://code.jquery.com/ui/1.9.0/themes/base/jquery-ui.css" />
<script src="http://code.jquery.com/jquery-1.8.2.js"></script>
<script src="http://code.jquery.com/ui/1.9.0/jquery-ui.js"></script>
<script>
$(function() {
    $('#datepicker').datepicker({ dateFormat: "yy-mm-dd" });
});
</script>
<script>
$(function() {
    $('#datepicker2').datepicker({ dateFormat: "yy-mm-dd" });
});
</script>
</head>
<body>
<div class="capcalera">
<table border="0" cellspacing="5" cellpadding="3" align="left">
<tr><td rowspan="3" align="left"><a href="index.jsp"></a></td><td
valign="bottom"><b>Benvingut al gestor d'informació</b></td></tr>
<tr><td valign="top"><b>del Jeroni de Moragas!</b></td></tr>
</table>
</div>
<div class="imatge">&nbsp;</div>

<div class="cos">
<b>Modificar professional</b><br><br>

<form action="./GestorProfessionals" method="post" enctype="multipart/form-data" class="form-horizontal" accept-
charset="utf-8" id="lomeuform" name="g_professionals">
<input type="radio" name="optionsRadios2" id="optionsRadios2" value="modificarex" checked hidden>
<div id="veuprof">
<div class="form-group">
<div class="col-sm-2">
Nom <input type="text" class="form-control" id="data" name="nom" size="15"
value="${professional.nom}">
1r cognom <input type="text" class="form-control" id="data" name="cognom1" size="15"
value="${professional.cognom1}">
2n cognom <input type="text" class="form-control" id="data" name="cognom2" size="15"
value="${professional.cognom2}">
</div><br>
<div class="col-sm-2">
NIF <input type="text" class="form-control" id="data" name="nifa" size="15" readonly
value="${professional.nif}">
Categoria
<select id="Select" class="form-control" name="categoria" style="width: 200px">
<c:forEach items="${categories}" var="option">
<option value="${option}"
${option == professional.nomcategoria ? 'selected' : ''}
>${option}</option>
</c:forEach>
</select>
Usuari actiu?
<select id="Select" class="form-control" name="actiu" style="width: 50px">
<c:forEach items="si,no" var="option14">
<option value="${option14}"
<c:forEach items="${professional.actiu}" var="option15">
${option14 == option15 ? 'selected' : ''}
</c:forEach>
>${option14}
</option>
</c:forEach>
</select>
</div><br>
Horari del professional:
<a href="${professional.lkhorari}">
<span id="abrevio">
</span>
<!--Script per eliminar rutes en mostrar els arxius-->

```

```

        <script>
            function Abrevio(argument) {
                var n = argument.lastIndexOf("/");
                var res = argument.substring(n+1, argument.length);
                return res;
            }
            document.getElementById("abrevio").innerHTML = Abrevio('${professional.lkhorari}');
        </script>
    </a>
    <input type='hidden' name='horariprof' value='${professional.lkhorari}'> <br><br>
    <div class="col-sm-2">
        Nou arxiu d'horaris del professional: <input type="file" name="uploadFile">
    </div><br>
    <div class="radio">
        <label>
            <input type="radio" name="optionsRadios1" id="optionsRadios1" value="si"
    ${professional.tutoria=="si"?checked:''}> Sí tutor
        </label>
        <label>
            <input type="radio" name="optionsRadios1" id="optionsRadios2" value="no"
    ${professional.tutoria=="no"?checked:''}> No Sí tutor
        </label>
    </div><br>
    </div>
    <button type="submit" name="submit" id="submit" value="Publicar" class="boto meu">Modifica</button>
</form>
</div>
</body>
</html>

```

NomsAules.jsp

```

<%--
Document : NomsAules
Created on : 20-mar-2015, 18:31:21
Author : Dr. Carles Torras Font
--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/functions" prefix="fn" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>Gestor d'informació del CEE Jeroni de Moragas</title>

        <!--<link href="css/bootstrap.min.css" rel="stylesheet"-->
        <link rel="stylesheet" href="css/estils.css">
        <script type="text/javascript" src="js/00jquery-1.11.1.js"></script>
        <script type="text/javascript" src="js/00amagar.js"></script>
    </head>
    <body>
        <div class="capcalera">
            <table border="0" cellspacing="5" cellpadding="3" align="left">
                <tr><td rowspan="3" align="left"><a href="index.jsp"></a></td><td
            valign="bottom"><b>Benvingut al gestor d'informació</b></td></tr>
                <tr><td valign="top"><b>del Jeroni de Moragas!</b></td></tr>
            </table>
        </div>
        <div class="imatge">&nbsp;</div>

        <div class="cos">
            <b>Gestionar Noms d'Aules</b><br><br>

            <form action="./GestorAules1" method="post" class="form-horizontal" accept-charset="utf-8" id="lomeuform"
            name="g_nomsaules">
                <input type="hidden" name="accio" id="accio" value="nomaules">
                <input type="radio" name="optionsRadios2" id="optionsRadios2" value="modificarex" checked hidden>
                <div id="veuprof">
                    <div class="form-group">
                        <div class="col-sm-2">
                            Nou nom d'aula <input type="text" class="form-control" id="data" name="aulafegir" size="15">
                            Nom d'aula existent
                            <select id="Select" class="form-control" name="aulaborrar" style="width: 200px">
                                <c:forEach items="{l1listanomaula}" var="option">
                                    <option value="{option}">${option}</option>
                                </c:forEach>
                            </select>
                        </div><br>
                        <div class="radio">
                            Selecciona: &nbsp;<input type="radio" name="optionsRadios2" id="optionsRadios1" value="afegir">&nbsp;<b>Afegir
                        </div>
                        <div class="radio">
                            <input type="radio" name="optionsRadios2" id="optionsRadios1" value="eliminar">&nbsp;<b>
            Eliminar
                        </div>
                    </div>
                </div>
                <button type="submit" name="submit" id="submit" value="Publicar" class="boto meu">Executa</button>
            </form>

```



```

</div>
</body>
</html>

```

Index.jsp

```

<%--
Document   : index
Created on : 20-mar-2015, 18:31:21
Author    : Dr. Carles Torras Font
--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/functions" prefix="fn" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<%@ page session="true" %>
<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>Sistema de Gestió del CEE Jeroni de Moragas</title>

<!--<link href="css/bootstrap.min.css" rel="stylesheet"-->
<link rel="stylesheet" href="css/estils.css">
<script type="text/javascript" src="js/00jquery-1.11.1.js"></script>
<script type="text/javascript" src="js/00amagar.js"></script>

<!-- Codi del upclick: només l'utilitzo per canviar foto d'usuari -->
<script type="text/javascript" src="js/upclick-min.js"></script>

<!-- Codi del datapicker -->
<link rel="stylesheet" href="http://code.jquery.com/ui/1.9.0/themes/base/jquery-ui.css" />
<script src="http://code.jquery.com/jquery-1.8.2.js"></script>
<script src="http://code.jquery.com/ui/1.9.0/jquery-ui.js"></script>

<!-- Codi de l'add-on time-picker al datePicker -->
<link rel="stylesheet" href="css/jquery-ui-timepicker-addon.css">
<script type="text/javascript" src="js/jquery-ui-timepicker-addon.js"></script>

<!-- Programació del datePicker -->
<script>
$(function() {
    $('#datepicker').datepicker({ dateFormat: "yy-mm-dd" });
});
</script>
<script>
$(function() {
    $('#datepicker2').datepicker({ dateFormat: "yy-mm-dd" });
});
</script>
<script>
$(function() {
    $('#datepicker3').datepicker({ dateFormat: "yy-mm-dd" });
});
</script>
<script>
$(function() {
    $('#datepicker4').datepicker({ dateFormat: "yy-mm-dd" });
});
</script>
<script>
$(function() {
    $('#datepicker5').datepicker({ dateFormat: "yy-mm-dd" });
});
</script>
<script>
$(function() {
    $('#datepicker6').datepicker({ dateFormat: "yy-mm-dd" });
});
</script>
<script>
$(function() {
    $('#datepicker7').datetimepicker({ dateFormat: "yy-mm-dd", timeFormat: "hh:mm:ss" });
});
</script>
<script>
$(function() {
    $('#datepicker8').datetimepicker({ dateFormat: "yy-mm-dd", timeFormat: "hh:mm:ss" });
});
</script>

<!-- Script del veure/amagar -->
<script type="text/javascript">
function toggle_visibility(id) {
    var e = document.getElementById(id);
    if(e.style.display == 'block')
        e.style.display = 'none';
    else
        e.style.display = 'block';
}
</script>
<script type="text/javascript">
function toggle_visibility2(id) {
    var e = document.getElementById(id);
    e.style.display = 'none';
}

```

```

</script>

<!-- Script AJAX de cerca de persones -->
<script type="text/javascript">
$(document).ready(function() {
    $('#cercaper').click(function() {
        $.ajax({
            type: "POST",
            dataType: "json",
            url: "CercaPer",
            data: "data="+$('#dadescerca').val(),
            success: function(response) {
                //alert("Pet!");
                var mida = response.llista.length;
                $(".rescerca").html(" ");
                $(".rescerca").append("<br><b>Resultats de la cerca</b><br>");
                for (var i = 0; i < mida; i++) {
                    $(".rescerca").append(
                        "Persona: " + response.llista[i] + "<br>"
                    );
                }
            }
        });
    });
});
</script>

<!-- Script AJAX per descarregar accions -->
<script type="text/javascript">
$(document).ready(function() {
    $('#descarregar').click(function() {
        $.ajax({
            type: "POST",
            url: "Descarrega",
            success: function(response) {
                alert(response);
                $('#resdes').html("<a href=\""+response+"\">Arxiu</a>");
            }
        });
    });
});
</script>

<!-- Script per explicar com han acabat les operacions -->
<script type="text/javascript">
window.onload = function() {
    try {
        var resultat = ${resultat};
        if (resultat === 1) {
            alert("InserciÃ³ correcte");
        } else if (resultat === 2) {
            alert("ModificaciÃ³ correcte");
        } else if (resultat === 3) {
            alert("EliminaciÃ³ correcte");
        } else if (resultat === 0) {
            alert("No s'ha pogut efectuar l'operaciÃ³");
        }
    } catch (error) {}
};
</script>

</head>
<body>
<div class="capcalera">
<table border="0" cellspacing="5" cellpadding="3" align="left">
<tr><td rowspan="3" align="left"><a href="index.jsp"></a></td><td
valign="bottom"><b>Benvingut al Sistema de Gestió</b></td></tr>
<tr><td valign="top"><b>del CEE Jeroni de Moragas!</b></td></tr>
</table>
</div>
<div class="imatge">&nbsp;</div>

<div class="cos">
<div class="avisos">
<table border="0" cellspacing="5" cellpadding="3" align="left">
<tr>
<td rowspan="3" align="left">

<td valign="bottom">
Benvingut ${usuariltot}. Has iniciat sessió correctament
</td>
</tr>
<tr>
<td valign="top">
<button type="button" class="boto2 meu" onclick="CanviPass()">Canviar la clau d'accÃ©s</button>
<script>
function CanviPass() {
    var clauvella = prompt("Introdueix la clau vella: ");
    var claunova1 = prompt("Introdueix la clau nova: ");
    var claunova2 = prompt("Confirma la clau nova: ");
    if (clauvella !== null && claunova1 !== null && claunova1 === claunova2) {
        $.ajax({
            type: "POST",
            url: "GestorClau",
            data: {"clauvella": clauvella, "claunova": claunova1},
            success: function(response) {
                var resultat = response;
                //alert(resultat);
            }
        });
    }
};
</script>
</td>
</tr>
</table>
</div>

```



```

checked> 
</label>
</label>
 No
</label>
</div><br>
</div>
<button type="submit" name="submit" id="submit" value="Publicar" class="boto
meu">Afegir</button>
</div>
</div>
</form>
</div>
<div class="cl">&nbsp;</div>
</div>
<div class="avisos">
<div class="trigger open"> <a href="#">Gestor d'alumnes</a></div>
<div class="cnt" style="width: 1000px">
<form action="/GestorAlumnes" ENCTYPE="multipart/form-data" class="form-horizontal" method="post"
accept-charset="utf-8" id="lomeuform" name="g_alumnes">
<div class="radio">
<button type="submit" name="submit" id="submit" value="Publicar" class="boto2
meu">Seleccionar</button>
<input type="radio" name="optionsRadios2" id="optionsRadios2" value="l·listar" checked
onclick="toggle_visibility2('veualumnes');">
&nbsp;&nbsp;&nbsp; L·listar, modificar o eliminar &nbsp;&nbsp;&nbsp; &nbsp;&nbsp;&nbsp;
</label>
<input type="radio" name="optionsRadios2" id="optionsRadios1" value="afegir"
onclick="toggle_visibility('veualumnes');">&nbsp;&nbsp;&nbsp; Afegir
</label>
<br><br>
<div id='veualumnes' style="display: none">
<div class="form-group" style="background-color: #f9faed">
<br><br>
<div class="col-sm-2">
Nom <input type="text" class="form-control" id="data" placeholder="Nom"
name="nom" size="15">
1r cognom <input type="text" class="form-control" id="data" placeholder="Primer
cognom" name="cognom1" size="15">
2n cognom <input type="text" class="form-control" id="data" placeholder="Primer
cognom" name="cognom2" size="15">
</div><br>
<div class="col-sm-2">
NIF <input type="text" class="form-control" id="data" placeholder="NIF"
name="nifa" size="15">
Data de naixement <input type="text" class="form-control" id="datepicker"
placeholder="Data" name="datanaix" size="6">
Any d'entrada <input type="text" class="form-control" id="data" placeholder="Any"
name="entrada" size="6">
</div><br>
<div class="col-sm-2">
AdreÀa <input type="text" class="form-control" id="adreÀa"
placeholder="adreÀa" name="adreca" size="35">
PoblaciÀ
<select id="Select" class="form-control" name="poblacio" style="width: 300px">
<c:forEach items="{municipis}" var="option">
<option value="{option}">{option}</option>
</c:forEach>
</select>
</div><br>
<div class="col-sm-2">
TelÀfon 1 <input type="text" class="form-control" id="tel1" placeholder="tel 1"
name="tel1" size="13">
TelÀfon 2 <input type="text" class="form-control" id="tel2" placeholder="tel 2"
name="tel2" size="13">
TelÀfon 3 <input type="text" class="form-control" id="tel3" placeholder="tel 3"
name="tel3" size="13">
TelÀfon 4 <input type="text" class="form-control" id="tel4" placeholder="tel 4"
name="tel4" size="13">
</div><br>
<div class="col-sm-2">
Nom del pare <input type="text" class="form-control" id="pare" placeholder="pare"
name="pare" size="13">
Nom de la mare <input type="text" class="form-control" id="mare"
placeholder="mare" name="mare" size="13">
Nom del tutor/a <input type="text" class="form-control" id="mare"
placeholder="tutor" name="nomtutor" size="13">
</div><br>
<div class="col-sm-2">
Nom de la parella del pare <input type="text" class="form-control" id="pare"
placeholder="parella pare" name="parellapare" size="13">
Nom de la parella de la mare <input type="text" class="form-control" id="mare"
placeholder="parella mare" name="parellamare" size="13">
</div><br>
<div class="col-sm-2">
NÀ de targeta sanità ria <input type="text" class="form-control" id="email"
placeholder="num targeta sanità ria" name="numtarsan" size="18">
Email <input type="text" class="form-control" id="email" placeholder="e-mail"
name="email" size="13">
</div><br>
<div class="col-sm-2">
DiagnÀstic 1 <input type="text" class="form-control" id="diag1" name="diag1"
size="90">
</div><br>
</div><br>

```



```

        </div>
    </div>
</div><br>
</form>
</div>
<div class="cl">&nbsp;&nbsp;&nbsp;</div>
</div>
<div class="avisos">
<div style="width: 1000px; height: 28px">
<form action="/GestorAules" ENCTYPE="multipart/form-data" class="form-horizontal" method="post" accept-
charset="utf-8" id="lomeuform" name="g_aules">
<input type="hidden" name="optionsRadios2" id="optionsRadios2" value="nomsaules" checked>
<button class="remove_button_css" type="submit" name="submit" id="submit">Gestor d'aules</button>
</form>
</div>
</div>
<div class="avisos">
<div class="trigger open"><a href="#">Gestor de normes</a></div>
<div class="cnt" style="width: 1000px">
<form action="/GestorNormes" ENCTYPE="multipart/form-data" class="form-horizontal" method="post"
accept-charset="utf-8" id="lomeuform" name="g_normes">
<div class="radio">
<button type="submit" name="submit" id="submit" value="Publicar" class="boto2
meu">Seleccionar</button>
<label>
<input type="radio" name="optionsRadios2" id="optionsRadios2" value="llistar" checked
onclick="toggle_visibility2('veunormes');">&nbsp;&nbsp;&nbsp;Llistar, modificar o eliminar
</label>
<label>
<input type="radio" name="optionsRadios2" id="optionsRadios1" value="afegir"
onclick="toggle_visibility('veunormes');">&nbsp;&nbsp;&nbsp;Afegir
</label>
<div id='veunormes' style="display: none">
<br><br>
<div class="form-group" style="background-color: #f9faed">
<br><br>
<div class="col-sm-2">
TÀ-tol <input type="text" class="form-control" id="titol" placeholder="TÀ-tol"
name="titol" size="75">
</div><br>
<div class="col-sm-2">
Arxiu explicatiu de la norma: <input type="file" name="uploadFile">
</div><br>
<button type="submit" name="submit" id="submit" value="Publicar" class="boto
meu">Afegir</button>
</div>
</div>
</div><br>
</form>
</div>
<div class="cl">&nbsp;&nbsp;&nbsp;</div>
</div>
<div class="avisos">
<button class="remove_button_css" type="submit" id="descarregar">Descarregar llistat d'accions: </button>
<div id="resdes" style="display: inline"></div>
</div><br><br><br>
</c:if>
<c:if test="{(sessionScope.rol == 'Mestre/a') || (rol == 'Logopeda') || (rol == 'Administrador/a') || (rol ==
'Fisioterapeuta') || (rol == 'Director/a')}">
<b style="color: #9EBCA0">Gestió del tutor</b><br><br>
<div class="avisos">
<div class="trigger open"><a href="#">Gestor d'assistència</a></div>
<div class="cnt" style="width: 1000px">
<form action="/GestorAssistencia" method="post" ENCTYPE="multipart/form-data" class="form-
horizontal" accept-charset="utf-8" id="lomeuform" name="g_assistencia">
<div class="radio">
<button type="submit" name="submit" id="submit" value="Publicar" class="boto2
meu">Seleccionar</button>
<label>
<input type="radio" name="optionsRadios2" id="optionsRadios2" value="llistar" checked
onclick="toggle_visibility2('assistencia');">&nbsp;&nbsp;&nbsp;Llistar o eliminar
</label>
<label>
<input type="radio" name="optionsRadios2" id="optionsRadios1" value="afegir"
onclick="toggle_visibility('assistencia');">&nbsp;&nbsp;&nbsp;Afegir
</label>
<div id='assistencia' style="display: none">
<br><br>
<div class="form-group" style="background-color: #f9faed">
<br>
<div class="col-sm-2">
Alumne
<select id="Select" class="form-control" name="alumnes" style="width: 300px">
<c:forEach items="{alumnesprofe}" var="option13">
<option value="{option13}">{option13}</option>
</c:forEach>
</select>
</div><br>
<div class="col-sm-2">
Motiu <input type="text" class="form-control" id="data" placeholder="motiu"
name="motiu" size="100">
</div><br>
<div class="col-sm-2">
Data <input type="text" class="form-control" id="datepicker3" placeholder="Data"
name="data" size="15">
</div><br>

```

Preindex.jsp

```
<%--
  Document   : Preindex
  Author    : Dr. Carles Torras Font
--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>Benvingut al CEE Jeroni de Moragas</title>
  </head>
  <body>
    <h1>Gestor d'informació del CEE Jeroni de Moragas</h1>
    <jsp:forward page="/Index">
      <jsp:param name="resultat" value="${resultat}"></jsp:param>
    </jsp:forward>
  </body>
</html>
```

Acces.java

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package beans;

/**
 *
 * @author Dr. Carles Torras Font
 */
public class Acces {
    private int idacces;
    private String nomaccio;
    private String nif;
    private String accesdh;

    public Acces(int idacces, String nomaccio, String nif, String accesdh) {
        this.nomaccio = nomaccio;
        this.idacces = idacces;
        this.nif = nif;
        this.accesdh = accesdh;
    }

    public int getIdacces() {
        return idacces;
    }

    public void setIdacces(int idacces) {
        this.idacces = idacces;
    }

    public String getNomaccio() {
        return nomaccio;
    }

    public void setNomaccio(String nomaccio) {
        this.nomaccio = nomaccio;
    }

    public String getNif() {
        return nif;
    }

    public void setNif(String nif) {
        this.nif = nif;
    }

    public String getAccesdh() {
        return accesdh;
    }

    public void setAccesdh(String accesdh) {
        this.accesdh = accesdh;
    }

    @Override
    public String toString() {
        return "Idacces=" + idacces;
    }
}
```

Accesals.java

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

package beans;

/**
 *
 * @author Dr. Carles Torras Font
 */
public class Accesals {
    private int idacces;
    private String nif;

    public Accesals(int idacces, String nif) {
        this.idacces = idacces;
        this.nif = nif;
    }

    public int getIdacces() {
        return idacces;
    }

    public void setIdacces(int idacces) {
        this.idacces = idacces;
    }

    public String getNif() {
        return nif;
    }

    public void setNif(String nif) {
        this.nif = nif;
    }

    @Override
    public String toString() {
        return "Idacces=" + idacces;
    }
}
```

Accio.java

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

package beans;

/**
 *
 * @author Dr. Carles Torras Font
 */
public class Accio {
    private String nomaccio;

    public Accio(String nomaccio) {
        this.nomaccio = nomaccio;
    }

    public String getNomaccio() {
        return nomaccio;
    }

    public void setNomaccio(String nomaccio) {
        this.nomaccio = nomaccio;
    }

    @Override
    public String toString() {
        return "NomAccio=" + nomaccio;
    }
}
```

Alumclasse.java

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

package beans;
```

```

/**
 *
 * @author Dr. Carles Torras Font
 */
public class Alumclasse {
    private String nomaula;
    private String nif;
    private String curs;

    public Alumclasse(String nomaula, String nif, String curs) {
        this.nomaula = nomaula;
        this.nif = nif;
        this.curs = curs;
    }

    public String getNomaula() {
        return nomaula;
    }

    public void setNomaula(String nomaula) {
        this.nomaula = nomaula;
    }

    public String getNif() {
        return nif;
    }

    public void setNif(String nif) {
        this.nif = nif;
    }

    public String getCurs() {
        return curs;
    }

    public void setCurs(String curs) {
        this.curs = curs;
    }

    @Override
    public String toString() {
        return "Nomaula: " + nomaula + " NIF: " + nif;
    }
}

```

Alumne.java

```

/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

package beans;

/**
 *
 * @author Dr. Carles Torras Font
 */
public class Alumne {
    private String nif;
    private String compartit;
    private String entrada;
    private String naixement;
    private String nompere;
    private String nommare;
    private String adreca;
    private String email;
    private String telefon1;
    private String telefon2;
    private String telefon3;
    private String telefon4;
    private String diag1;
    private String diag2;
    private String diag3;
    private String diag4;
    private String diag5;
    private String med1;
    private String med2;
    private String med3;
    private String med4;
    private String med5;
    private int idmun;
    private String nom;
    private String cognom1;
    private String cognom2;
    private String programa;
    private String numtarsan;
    private String nomparellapare;
    private String nomparellamare;
    private String nomtutortutora;

    public Alumne(String nif, String compartit, String entrada, String naixement,
        String nompere, String nommare, String adreca, String email,
        String telefon1, String telefon2, String telefon3, String telefon4,
        String diag1, String diag2, String diag3, String diag4, String diag5,

```

```
String med1, String med2, String med3, String med4, String med5, int idmun,
String nom, String cognom1, String cognom2, String programa,
String numtarsan, String nomparellapare, String nomparellamare,
String nomtutortutora) {

    this.nif = nif;
    this.compartit = compartit;
    this.entrada = entrada;
    this.naixement = naixement;
    this.nompare = nompare;
    this.nommare = nommare;
    this.adreca = adreca;
    this.email = email;
    this.telefon1 = telefon1;
    this.telefon2 = telefon2;
    this.telefon3 = telefon3;
    this.telefon4 = telefon4;
    this.diag1 = diag1;
    this.diag2 = diag2;
    this.diag3 = diag3;
    this.diag4 = diag4;
    this.diag5 = diag5;
    this.med1 = med1;
    this.med2 = med2;
    this.med3 = med3;
    this.med4 = med4;
    this.med5 = med5;
    this.idmun = idmun;
    this.nom = nom;
    this.cognom1 = cognom1;
    this.cognom2 = cognom2;
    this.programa = programa;
    this.numtarsan = numtarsan;
    this.nomparellapare = nomparellapare;
    this.nomparellamare = nomparellamare;
    this.nomtutortutora = nomtutortutora;
}

public String getNif() {
    return nif;
}

public void setNif(String nif) {
    this.nif = nif;
}

public String getCompartit() {
    return compartit;
}

public void setCompartit(String compartit) {
    this.compartit = compartit;
}

public String getEntrada() {
    return entrada;
}

public void setEntrada(String entrada) {
    this.entrada = entrada;
}

public String getNaixement() {
    return naixement;
}

public void setNaixement(String naixement) {
    this.naixement = naixement;
}

public String getNompare() {
    return nompare;
}

public void setNompare(String nompare) {
    this.nompare = nompare;
}

public String getNommare() {
    return nommare;
}

public void setNommare(String nommare) {
    this.nommare = nommare;
}

public String getAdreca() {
    return adreca;
}

public void setAdreca(String adreca) {
    this.adreca = adreca;
}

public String getEmail() {
    return email;
}

public void setEmail(String email) {
```

```
        this.email = email;
    }

    public String getTelefon1() {
        return telefon1;
    }

    public void setTelefon1(String telefon1) {
        this.telefon1 = telefon1;
    }

    public String getTelefon2() {
        return telefon2;
    }

    public void setTelefon2(String telefon2) {
        this.telefon2 = telefon2;
    }

    public String getTelefon3() {
        return telefon3;
    }

    public void setTelefon3(String telefon3) {
        this.telefon3 = telefon3;
    }

    public String getTelefon4() {
        return telefon4;
    }

    public void setTelefon4(String telefon4) {
        this.telefon4 = telefon4;
    }

    public String getDiag1() {
        return diag1;
    }

    public void setDiag1(String diag1) {
        this.diag1 = diag1;
    }

    public String getDiag2() {
        return diag2;
    }

    public void setDiag2(String diag2) {
        this.diag2 = diag2;
    }

    public String getDiag3() {
        return diag3;
    }

    public void setDiag3(String diag3) {
        this.diag3 = diag3;
    }

    public String getDiag4() {
        return diag4;
    }

    public void setDiag4(String diag4) {
        this.diag4 = diag4;
    }

    public String getDiag5() {
        return diag5;
    }

    public void setDiag5(String diag5) {
        this.diag5 = diag5;
    }

    public String getMed1() {
        return med1;
    }

    public void setMed1(String med1) {
        this.med1 = med1;
    }

    public String getMed2() {
        return med2;
    }

    public void setMed2(String med2) {
        this.med2 = med2;
    }

    public String getMed3() {
        return med3;
    }

    public void setMed3(String med3) {
        this.med3 = med3;
    }
}
```

```
public String getMed4() {
    return med4;
}

public void setMed4(String med4) {
    this.med4 = med4;
}

public String getMed5() {
    return med5;
}

public void setMed5(String med5) {
    this.med5 = med5;
}

public int getIdmun() {
    return idmun;
}

public void setIdmun(int idmun) {
    this.idmun = idmun;
}

public String getNom() {
    return nom;
}

public void setNom(String nom) {
    this.nom = nom;
}

public String getCognom1() {
    return cognom1;
}

public void setCognom1(String cognom1) {
    this.cognom1 = cognom1;
}

public String getCognom2() {
    return cognom2;
}

public void setCognom2(String cognom2) {
    this.cognom2 = cognom2;
}

public String getPrograma() {
    return programa;
}

public void setPrograma(String programa) {
    this.programa = programa;
}

public String getNomparellapare() {
    return nonparellapare;
}

public void setNomparellapare(String nonparellapare) {
    this.nonparellapare = nonparellapare;
}

public String getNumtarsan() {
    return numtarsan;
}

public void setNumtarsan(String numtarsan) {
    this.numtarsan = numtarsan;
}

public String getNomparellamare() {
    return nonparellamare;
}

public void setNomparellamare(String nonparellamare) {
    this.nonparellamare = nonparellamare;
}

public String getNomtutortutora() {
    return nomtutortutora;
}

public void setNomtutortutora(String nomtutortutora) {
    this.nomtutortutora = nomtutortutora;
}

@Override
public String toString() {
    return nom + " " + cognom1 + ", " + niF;
}
}
```

Assistencia.java

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

package beans;

/**
 *
 * @author Dr. Carles Torras Font
 */
public class Assistencia {
    private static final long serialVersionUID = 1L;
    private Integer idassistencia;
    private String data;
    private String motiu;
    private String professional;
    private String alumne;

    public Assistencia(Integer idassistencia, String data, String motiu, String alumne, String professional) {
        this.idassistencia = idassistencia;
        this.data = data;
        this.motiu = motiu;
        this.alumne = alumne;
        this.professional = professional;
    }

    public Integer getIdassistencia() {
        return idassistencia;
    }

    public void setIdassistencia(Integer idassistencia) {
        this.idassistencia = idassistencia;
    }

    public String getData() {
        return data;
    }

    public void setData(String data) {
        this.data = data;
    }

    public String getMotiu() {
        return motiu;
    }

    public void setMotiu(String motiu) {
        this.motiu = motiu;
    }

    public String getAlumne() {
        return alumne;
    }

    public void setAlumne(String alumne) {
        this.alumne = alumne;
    }

    public String getProfessional() {
        return professional;
    }

    public void setProfessional(String professional) {
        this.professional = professional;
    }

    @Override
    public String toString() {
        return "Assistencia ID:" + idassistencia;
    }
}
```

Aula.java

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

package beans;

/**
 *
 * @author Dr. Carles Torras Font
 */
public class Aula {
    private String nomaula;

    public Aula(String nomaula) {
```



```
        this.nomaula = nomaula;
    }

    public String getNomaula() {
        return nomaula;
    }

    public void setNomaula(String nomaula) {
        this.nomaula = nomaula;
    }

    @Override
    public String toString() {
        return "Nomaula=" + nomaula;
    }
}
```

Categoria.java

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

package beans;

/**
 *
 * @author Dr. Carles Torras Font
 */
public class Categoria {
    private String nom;

    public Categoria(String nom) {
        this.nom = nom;
    }

    public String getNom() {
        return nom;
    }

    public void setNom(String nom) {
        this.nom = nom;
    }

    @Override
    public String toString() {
        return "Nom de categoria:" + nom;
    }
}
```

Classes.java

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

package beans;

/**
 *
 * @author Dr. Carles Torras Font
 */
public class Classes {
    private String nomaula;
    private String curs;
    private String lkhorari;

    public Classes(String nomaula, String curs, String lkhorari) {
        this.lkhorari = lkhorari;
        this.nomaula = nomaula;
        this.curs = curs;
    }

    public String getNomaula() {
        return nomaula;
    }

    public void setNomaula(String nomaula) {
        this.nomaula = nomaula;
    }

    public String getCurs() {
        return curs;
    }

    public void setCurs(String curs) {
        this.curs = curs;
    }
}
```

```
public String getLkhorari() {
    return lkhorari;
}

public void setLkhorari(String lkhorari) {
    this.lkhorari = lkhorari;
}

@Override
public String toString() {
    return nomaula + ", " + curs;
}
}
```

Cursos.java

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

package beans;

/**
 *
 * @author Dr. Carles Torras Font
 */
public class Cursos {
    private String curs;

    public Cursos(String curs) {
        this.curs = curs;
    }

    public String getCurs() {
        return curs;
    }

    public void setCurs(String curs) {
        this.curs = curs;
    }

    @Override
    public String toString() {
        return "Curs:" + curs;
    }
}
```

Fotoalumne.java

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

package beans;

/**
 *
 * @author Dr. Carles Torras Font
 */
public class Fotoalumne {
    private Integer id;
    private String nif;

    public Fotoalumne(Integer id, String nif) {
        this.id = id;
        this.nif = nif;
    }

    public Integer getId() {
        return id;
    }

    public void setId(Integer id) {
        this.id = id;
    }

    public String getNif() {
        return nif;
    }

    public void setNif(String nif) {
        this.nif = nif;
    }

    @Override
    public String toString() {
        return "Id: " + id;
    }
}
```

Fotografia.java

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

package beans;

/**
 *
 * @author Dr. Carles Torras Font
 */
public class Fotografia {
    private Integer id;
    private String data;
    private String nomaula;
    private String curs;
    private String lkfoto;
    private String etiquetal;
    private String etiqueta2;
    private String etiqueta3;
    private String etiqueta4;

    public Fotografia(Integer id, String data, String nomaula, String curs, String lkfoto,
        String etiquetal, String etiqueta2, String etiqueta3, String etiqueta4) {
        this.id = id;
        this.data = data;
        this.nomaula = nomaula;
        this.curs = curs;
        this.lkfoto = lkfoto;
        this.etiquetal = etiquetal;
        this.etiqueta2 = etiqueta2;
        this.etiqueta3 = etiqueta3;
        this.etiqueta4 = etiqueta4;
    }

    public Integer getId() {
        return id;
    }

    public void setId(Integer id) {
        this.id = id;
    }

    public String getData() {
        return data;
    }

    public void setData(String data) {
        this.data = data;
    }

    public String getNomaula() {
        return nomaula;
    }

    public void setNomaula(String nomaula) {
        this.nomaula = nomaula;
    }

    public String getCurs() {
        return curs;
    }

    public void setCurs(String curs) {
        this.curs = curs;
    }

    public String getLkfoto() {
        return lkfoto;
    }

    public void setLkfoto(String lkfoto) {
        this.lkfoto = lkfoto;
    }

    public String getEtiquetal() {
        return etiquetal;
    }

    public void setEtiquetal(String etiquetal) {
        this.etiquetal = etiquetal;
    }

    public String getEtiqueta2() {
        return etiqueta2;
    }

    public void setEtiqueta2(String etiqueta2) {
        this.etiqueta2 = etiqueta2;
    }

    public String getEtiqueta3() {
```

```
        return etiqueta3;
    }

    public void setEtiqueta3(String etiqueta3) {
        this.etiqueta3 = etiqueta3;
    }

    public String getEtiqueta4() {
        return etiqueta4;
    }

    public void setEtiqueta4(String etiqueta4) {
        this.etiqueta4 = etiqueta4;
    }

    @Override
    public String toString() {
        return "Id: " + id;
    }
}
```

Incialumne.java

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

package beans;

/**
 *
 * @author Dr. Carles Torras Font
 */
public class Incialumne {
    private int idincidencia;
    private String nif;

    public Incialumne(int idincidencia, String nif) {
        this.idincidencia = idincidencia;
        this.nif = nif;
    }

    public int getIdincidencia() {
        return idincidencia;
    }

    public void setIdincidencia(int idincidencia) {
        this.idincidencia = idincidencia;
    }

    public String getNif() {
        return nif;
    }

    public void setNif(String nif) {
        this.nif = nif;
    }

    @Override
    public String toString() {
        return "Idincidencia: " + idincidencia + " NIF: " + nif;
    }
}
```

Incidencia.java

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

package beans;

/**
 *
 * @author Dr. Carles Torras Font
 */
public class Incidencia {
    private Integer idincidencia;
    private String lkinforme;
    private String data;
    private String titol;
    private String nif;

    public Incidencia(Integer idincidencia, String data, String titol, String nif, String lkinforme) {
        this.idincidencia = idincidencia;
        this.data = data;
        this.titol = titol;
        this.nif = nif;
        this.lkinforme = lkinforme;
    }
}
```

```

    }

    public Integer getIdincidencia() {
        return idincidencia;
    }

    public void setIdincidencia(Integer idincidencia) {
        this.idincidencia = idincidencia;
    }

    public String getLkinforme() {
        return lkinforme;
    }

    public void setLkinforme(String lkinforme) {
        this.lkinforme = lkinforme;
    }

    public String getData() {
        return data;
    }

    public void setData(String data) {
        this.data = data;
    }

    public String getTitol() {
        return titol;
    }

    public void setTitol(String titol) {
        this.titol = titol;
    }

    public String getNif() {
        return nif;
    }

    public void setNif(String nif) {
        this.nif = nif;
    }

    @Override
    public String toString() {
        return "Idincidencia: " + idincidencia;
    }
}

```

Informe.java

```

/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

package beans;

/**
 *
 * @author Dr. Carles Torras Font
 */
public class Informe {
    private Integer idinforme;
    private String data;
    private String titolinforme;
    private String lkinforme;
    private String alumne;

    public Informe(Integer idinforme, String data, String titolinforme, String lkinforme, String alumne) {
        this.idinforme = idinforme;
        this.data = data;
        this.titolinforme = titolinforme;
        this.lkinforme = lkinforme;
        this.alumne = alumne;
    }

    public Integer getIdinforme() {
        return idinforme;
    }

    public void setIdinforme(Integer idinforme) {
        this.idinforme = idinforme;
    }

    public String getData() {
        return data;
    }

    public void setData(String data) {
        this.data = data;
    }

    public String getTitolinforme() {
        return titolinforme;
    }
}

```

```

public void setTitolinforme(String titolinforme) {
    this.titolinforme = titolinforme;
}

public String getLkinforme() {
    return lkinforme;
}

public void setLkinforme(String lkinforme) {
    this.lkinforme = lkinforme;
}

public String getAlumne() {
    return alumne;
}

public void setAlumne(String alumne) {
    this.alumne = alumne;
}

@Override
public String toString() {
    return "Idinforme: " + idinforme;
}
}

```

Municipi.java

```

/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

package beans;

/**
 *
 * @author Dr. Carles Torras Font
 */
public class Municipi {
    private String nommunicipi;
    private String nomprovincia;
    private int idmun;

    public Municipi(int idmun, String nommunicipi, String nomprovincia) {
        this.nommunicipi = nommunicipi;
        this.nomprovincia = nomprovincia;
        this.idmun = idmun;
    }

    public String getNommunicipi() {
        return nommunicipi;
    }

    public void setNommunicipi(String nommunicipi) {
        this.nommunicipi = nommunicipi;
    }

    public String getNomprovincia() {
        return nomprovincia;
    }

    public void setNomprovincia(String nomprovincia) {
        this.nomprovincia = nomprovincia;
    }

    public int getIdmun() {
        return idmun;
    }

    public void setIdmun(int idmun) {
        this.idmun = idmun;
    }

    @Override
    public String toString() {
        return "Nom del municipi:" + nommunicipi;
    }
}

```

Normes.java

```

/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

package beans;

/**
 *
 * @author Dr. Carles Torras Font
 */

```

```
*/
public class Normes {
    private String idnormes;
    private String titolnorma;
    private String lknorma;

    public Normes(String idnormes, String titolnorma, String lknorma) {
        this.idnormes = idnormes;
        this.titolnorma = titolnorma;
        this.lknorma = lknorma;
    }

    public String getIdnormes() {
        return idnormes;
    }

    public void setIdnormes(String idnormes) {
        this.idnormes = idnormes;
    }

    public String getTitolnorma() {
        return titolnorma;
    }

    public void setTitolnorma(String titolnorma) {
        this.titolnorma = titolnorma;
    }

    public String getLknorma() {
        return lknorma;
    }

    public void setLknorma(String lknorma) {
        this.lknorma = lknorma;
    }

    @Override
    public String toString() {
        return "Idnormes: " + idnormes;
    }
}
}
```

Profclasse.java

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

package beans;

/**
 *
 * @author Dr. Carles Torras Font
 */
public class Profclasse {
    private String nomaula;
    private String nifprof;
    private String curs;

    public Profclasse(String nomaula, String nifprof, String curs) {
        this.nomaula = nomaula;
        this.nifprof = nifprof;
        this.curs = curs;
    }

    public String getNomaula() {
        return nomaula;
    }

    public void setNomaula(String nomaula) {
        this.nomaula = nomaula;
    }

    public String getNifprof() {
        return nifprof;
    }

    public void setNifprof(String nifprof) {
        this.nifprof = nifprof;
    }

    public String getCurs() {
        return curs;
    }

    public void setCurs(String curs) {
        this.curs = curs;
    }

    @Override
    public String toString() {
        return "Nomaula: " + nomaula + " NIF: " + nifprof;
    }
}
}
```

Professional.java

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

package beans;

/**
 *
 * @author Dr. Carles Torras Font
 */
public class Professional {
    private static final long serialVersionUID = 1L;
    public String nif;
    private String tutoria;
    private String lkhorari;
    private String nom;
    private String cognom1;
    private String cognom2;
    private String password;
    private String nomcategoria;
    private String lkfotousuari;
    private String actiu;

    public Professional(String nif, String tutoria, String lkhorari, String nom,
        String cognom1, String cognom2, String password, String nomcategoria,
        String lkfotousuari, String actiu) {
        this.nif = nif;
        this.tutoria = tutoria;
        this.lkhorari = lkhorari;
        this.nom = nom;
        this.cognom1 = cognom1;
        this.cognom2 = cognom2;
        this.password = password;
        this.nomcategoria = nomcategoria;
        this.lkfotousuari = lkfotousuari;
        this.actiu = actiu;
    }

    public String getNif() {
        return nif;
    }

    public void setNif(String nif) {
        this.nif = nif;
    }

    public String getTutoria() {
        return tutoria;
    }

    public void setTutoria(String tutoria) {
        this.tutoria = tutoria;
    }

    public String getLkhorari() {
        return lkhorari;
    }

    public void setLkhorari(String lkhorari) {
        this.lkhorari = lkhorari;
    }

    public String getNom() {
        return nom;
    }

    public void setNom(String nom) {
        this.nom = nom;
    }

    public String getCognom1() {
        return cognom1;
    }

    public void setCognom1(String cognom1) {
        this.cognom1 = cognom1;
    }

    public String getCognom2() {
        return cognom2;
    }

    public void setCognom2(String cognom2) {
        this.cognom2 = cognom2;
    }

    public String getPassword() {
        return password;
    }

    public void setPassword(String password) {
```



```

        this.password = password;
    }

    public String getNomcategoria() {
        return nomcategoria;
    }

    public void setNomcategoria(String nomcategoria) {
        this.nomcategoria = nomcategoria;
    }

    public String getLkfotousuari() {
        return lkfotousuari;
    }

    public void setLkfotousuari(String lkfotousuari) {
        this.lkfotousuari = lkfotousuari;
    }

    public String getActiu() {
        return actiu;
    }

    public void setActiu(String actiu) {
        this.actiu = actiu;
    }

    @Override
    public String toString() {
        return nom + " "+cognoml+", " + nif;
    }
}

```

Profinformes.java

```

/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

package beans;

/**
 *
 * @author Dr. Carles Torras Font
 */
public class Profinformes {
    private int idinforme;
    private String nif;

    public Profinformes(int idinforme, String nif) {
        this.idinforme = idinforme;
        this.nif = nif;
    }

    public int getIdinforme() {
        return idinforme;
    }

    public void setIdinforme(int idinforme) {
        this.idinforme = idinforme;
    }

    public String getNif() {
        return nif;
    }

    public void setNif(String nif) {
        this.nif = nif;
    }

    @Override
    public String toString() {
        return "Idinforme: " + idinforme + " NIF: " + nif;
    }
}

```

Profreunions.java

```

/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

package beans;

/**
 *
 * @author Dr. Carles Torras Font
 */

```

```
public class Profreunions {
    private int idreunio;
    private String nif;

    public Profreunions(String nif, int idreunio) {
        this.idreunio = idreunio;
        this.nif = nif;
    }

    public int getIdreunio() {
        return idreunio;
    }

    public void setIdreunio(int idreunio) {
        this.idreunio = idreunio;
    }

    public String getNif() {
        return nif;
    }

    public void setNif(String nif) {
        this.nif = nif;
    }

    @Override
    public String toString() {
        return "Idreunio: " + idreunio + " NIF: " + nif;
    }
}
```

Reunio.java

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

package beans;

/**
 *
 * @author Dr. Carles Torras Font
 */
public class Reunio {
    private Integer idreunio;
    private String datahora;
    private String assistents;
    private String lkcontingut;
    private Integer propera;
    private String nif;

    public Reunio(Integer idreunio, String datahora, String assistents,
        String lkcontingut, Integer propera, String nif) {
        this.idreunio = idreunio;
        this.datahora = datahora;
        this.assistents = assistents;
        this.lkcontingut = lkcontingut;
        this.propera = propera;
        this.nif = nif;
    }

    public Integer getIdreunio() {
        return idreunio;
    }

    public void setIdreunio(Integer idreunio) {
        this.idreunio = idreunio;
    }

    public String getDatahora() {
        return datahora;
    }

    public void setDatahora(String datahora) {
        this.datahora = datahora;
    }

    public String getAssistents() {
        return assistents;
    }

    public void setAssistents(String assistents) {
        this.assistents = assistents;
    }

    public String getLkcontingut() {
        return lkcontingut;
    }

    public void setLkcontingut(String lkcontingut) {
        this.lkcontingut = lkcontingut;
    }
}
```

```

public Integer getPropera() {
    return propera;
}

public void setPropera(Integer propera) {
    this.propera = propera;
}

public String getNif() {
    return nif;
}

public void setNif(String nif) {
    this.nif = nif;
}

@Override
public String toString() {
    return "Idreunio=" + idreunio;
}
}

```

Connexio.java

```

package jdbc;

import java.sql.*;

public class Connexio {
    protected static Connection conn=null;
    private static String url="jdbc:mysql://localhost:3306/jeroni?user=jeroni&password=jeroni";

    public static Connection abrirConexion () {
        try {
            Class.forName("com.mysql.jdbc.Driver").newInstance();
            conn=DriverManager.getConnection(url);
        } catch (InstantiationException e) {
            e.printStackTrace();
        } catch (IllegalAccessException e) {
            e.printStackTrace();
        } catch (ClassNotFoundException e) {
            e.printStackTrace();
        } catch (SQLException e) {
            e.printStackTrace();
        }
        return conn;
    }

    public static void cerrarConexion(){
        if(conn!=null){
            try {
                conn.close();
            } catch (SQLException e) {
                e.printStackTrace();
            }
        }
    }
}

```

Interface.java

```

/*
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */
package jdbc;
import beans.Alumne;
import beans.Assistencia;
import beans.Aula;
import beans.Classes;
import beans.Fotoalumne;
import beans.Fotografia;
import beans.Incialumne;
import beans.Incidencia;
import beans.Informe;
import beans.Municipi;
import beans.Normes;
import beans.Professional;
import beans.Profinformes;
import beans.Profreunions;
import beans.Reunio;
import java.util.List;
/**
 *
 * @author Dr. Carles Torras Font
 */
public interface Interface {

    // Cerca gent
    public List <String> cercaGent(String string);

    // Professionals
    public int crearProfessional(Professional professional);
    public int modificarProfessional(String nif, String tutor, String arxiu,
        String nom, String cognom1, String cognom2, String categoria, String actiu);
}

```

```
public int canviPassword(String nif, String clau);
public Professional cercaProfessional(String nif);
public int eliminarProfessional(String nif);
public List <Professional> cercaProf();
public List <String> llistarTotsProfs();
public List <String> llistarTutors();
public List <String> Autenticacio(String usuari);
public int canviarFotoUsuari(String usuari, String arxiu);

// Alumnes
public int crearAlumne(Alumne alumne);
public int modificarAlumne(Alumne alumne);
public Alumne cercaAlumne(String nif);
public int eliminarAlumne(String nif);
public List <Alumne> cercaAlumne();
public List <String> llistarTotsAlumnes();
public List <String> NifAlumnesProf(String nif);

// AlumClasse
public int insertarAlumnesClasse(String nomaula, String curs, List<String> nif);
public int eliminarAlumnesClasse(String nomaula, String curs);
public List <String> llistarAlumnesClasse(String nomaula, String curs);

// ProfeClasse
public int insertarProfesClasse(String elnomaula, String elcursaula, List<String> nifprof);
public int eliminarProfesClasse(String elnomaula, String elcursaula);
public List <String> llistarProfesClasse(String elnomaula, String elcursaula);

// Classes
public int crearClasse(Classe classe);
public int eliminarClasse(String nom, String curs);
public Classe cercaClasse(String nom, String curs);
public int modificarClasse(Classe classe);
public List <Classe> llistarClasses();
public List <String> llistarClasseCurs();

// Aules i cursos
public int crearNomAules(Aula aules);
public int eliminarNomAules(String nom);
public List <String> llistarNomAules();
public List <String> llistarCursos();

// Municipis
public int cercarMunicipi(String nom);
public List <String> llistarMunicipis();
public List <Municipi> llistarTotMunicipis();

// Categories
public List <String> llistarCategories();

// Normes
public int crearNorma(String titol, String arxiu);
public Normes cercaNorma(String id);
public int modificarNorma(Normes norma);
public int eliminarNorma(String id);
public List <Normes> llistarNormes();
public List <String> llistarIdNormes();

// Assistència
public int crearAssistencia(String data, String motiu, String alumne, String professional);
public int eliminarAssistencia(String id);
public List <String> llistarIdAssistencia(String nif);
public List <Assistencia> llistarAssistencia(String nif);

// Incidència
public int crearIncidencia(String professional, String lkinforme, String data, String titol);
public int eliminarIncidencia(String id);
public List <String> llistarIdIncidencia(String nif);
public List <Incidencia> llistarIncidencia(String nif);

// Incialumne
public int crearIncialumne(String numinci, List<String> alumnes);
public int eliminarIncialumne(String id);
public List <Incialumne> llistarIncialumne(String nif);

// Informes
public int crearInforme(String alumne, String lkinforme, String data, String titol);
public int eliminarInforme(String id);
public int modificarInforme(String alumne, String lkinforme, String data, String titol, String id);
public List <String> llistarIdInformes(String nif);
public List <Informe> llistarInformes(String nif);
public Informe cercaInforme(String id);

// Profinformes
public int crearProfinformes(String numinf, List<String> professional);
public int eliminarProfinformes(String id);
public int modificarProfinformes(String numinf, List<String> professional);
public List <Profinformes> llistarProfinformes();
public List <String> llistarProfesInformes(String id);

// Reunions
public int crearReunio(String datahora, String assistents, String lkcontingut,
    String propera, String alumne);
public int eliminarReunio(String id);
public int modificarReunio(String datahora, String assistents, String lkcontingut,
    String propera, String alumne, String idreunio);
public List <String> llistarIdReunions(String nif);
public List <Reunio> llistarReunions(String nif);
public Reunio cercaReunio(String id);
```

```

// Profreunions
public int crearProfReunions(String idreunio, List<String> professional);
public int eliminarProfReunions(String idreunio);
public List <Profreunions> llistarProfReunions();
public List <String> llistarProfReunions(String id);

// Fotografies
public int crearFoto(String data, String nomaula, String curs, String lkfoto, String etiquetal,
String etiqueta2, String etiqueta3, String etiqueta4);
public int eliminarFoto(String id);
public int modificarFoto(String data, String nomaula, String curs, String lkfoto, String
etiquetal, String etiqueta2, String etiqueta3, String etiqueta4, String id);
public List <String> llistarIdFotos();
public List <Fotografia> llistarFotos();
public List <Fotografia> llistarFotosEspecial(String etl, String datal,
String data2, String aula_nom, String aula_curs, String nif);
public Fotografia cercaFoto(String id);

// Fotoalumni
public int crearFotoalumni(String foto, List<String> nif);
public int eliminarFotoalumni(String id);
public List <Fotoalumni> llistarFotoalumnis();
public List <String> cercaFotoalumnis(String id);
}

```

InterfaceRegistre.java

```

/*
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */
package jdbc;

import beans.Acces;
import beans.Accesals;
import java.util.List;

/**
 *
 * @author Dr. Carles Torras Font
 */
public interface InterfaceRegistre {

    public int nouRegistre(String accio, String nif, List<String> nifs);
    public List<String> llistarRegistre();
    public List<Acces> llistarTotRegistre();
    public List<Accesals> llistarAccesAl();

}

```

OpRegistre.java

```

package jdbc;

import beans.Acces;
import beans.Accesals;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.text.DateFormat;
import java.text.SimpleDateFormat;
import java.util.ArrayList;
import java.util.Date;
import java.util.List;

/**
 *
 * @author Dr. Carles Torras Font
 */
public class OpRegistre extends Connexio implements InterfaceRegistre {

    private static final String INSERTAR="INSERT INTO jeroni.acces (nif,nomaccio,accesdh) VALUES(?,?,?);";
    private static final String LLISTAR="SELECT * FROM jeroni.acces;";
    private static final String NUM_INSERTAR="SELECT MAX(idacces) AS maxim FROM jeroni.acces;";
    private static final String INSERTAR_ACCESALS="INSERT INTO jeroni.accesals (idaccesals,nifalumni) VALUES(?,?);";
    private static final String LLISTAR_ACCESAL="SELECT * FROM jeroni.accesals;";
    private PreparedStatement ps=null;

    @Override
    public int nouRegistre(String accio, String nif, List<String> nifs) {
        abrirConexion ();
        int t=0;

        // Creo el timestamp
        DateFormat format = new SimpleDateFormat("yyyy-MM-dd HH:mm:ss");
        Date diahora = new Date();
        String ara = format.format(diahora);

        try {
            ps = conn.prepareStatement(INSERTAR);
            ps.setString(1, nif);
            ps.setString(2, accio);

```

```

        ps.setString(3, ara);
        ps.executeUpdate();

        if(nifs!=null && !nifs.isEmpty()){
            try {
                if (nifs.size() > 0) {
                    // Pillo el nÃºmero de incidència que he creat
                    ps = conn.prepareStatement(NUM_INSERTIO);
                    ResultSet rs = ps.executeQuery();
                    while(rs.next()){
                        t = Integer.parseInt(rs.getString("maxim"));
                    }
                    ps=conn.prepareStatement(INSERTAR_ACCESALS);
                    for (int i = 0; i < nifs.size(); i++) {
                        String lonif = nifs.get(i);
                        if (lonif != null) {
                            ps.setInt(1, t);
                            ps.setString(2, lonif);
                            ps.addBatch();
                        }
                    }
                    ps.executeBatch();
                }
            } catch (NullPointerException e) {}
        }
    } catch (SQLException e) {
        e.printStackTrace();
    }
    cerrarConexion();
    return t;
}

@Override
public List<String> llistarRegistre() {
    List<String> llista = new ArrayList();
    String pel=null;
    abrirConexion();
    try {
        ps=conn.prepareStatement(LLISTAR);
        ResultSet rs =ps.executeQuery();
        while(rs.next()){
            pel="Id accÃs: "+rs.getString("idaces")+", NIF: "+rs.getString("nif")+", AcciÃ:
+rs.getString("nomaccio")+", NIF: "+rs.getString("acesdh");
            llista.add(pel);
        }
    } catch (SQLException e) {
        e.printStackTrace();
    }
    cerrarConexion();
    return llista;
}

@Override
public List<Accesals> llistarAccesAl() {
    List<Accesals> llista = new ArrayList();
    Accesals pel=null;
    abrirConexion();
    try {
        ps=conn.prepareStatement(LLISTAR_ACCESAL);
        ResultSet rs =ps.executeQuery();
        while(rs.next()){
            pel = new Accesals(0,null);
            pel.setIdaces(rs.getInt("idacesals"));
            pel.setNif(rs.getString("nifalume"));
            llista.add(pel);
        }
    } catch (SQLException e) {
        e.printStackTrace();
    }
    cerrarConexion();
    return llista;
}

@Override
public List<Acces> llistarTotRegistre() {
    List<Acces> llista = new ArrayList();
    Acces pel=null;
    abrirConexion();
    try {
        ps=conn.prepareStatement(LLISTAR);
        ResultSet rs =ps.executeQuery();
        while(rs.next()){
            pel = new Acces(0, null, null, null);
            pel.setIdaces(rs.getInt("idaces"));
            pel.setNomaccio(rs.getString("nomaccio"));
            pel.setNif(rs.getString("nif"));
            pel.setAccesdh(rs.getString("acesdh"));
            llista.add(pel);
        }
    } catch (SQLException e) {
        e.printStackTrace();
    }
    cerrarConexion();
    return llista;
}
}
}

```

OperacionsBD.java

```

package jdbc;

import beans.Alumne;
import beans.Assistencia;
import beans.Aula;
import beans.Classes;
import beans.Fotoalumne;
import beans.Fotografia;
import beans.Incialumne;
import beans.Incidencia;
import beans.Informe;
import beans.Municipi;
import beans.Normes;
import beans.Professional;
import beans.Profinformes;
import beans.Profreunions;
import beans.Reunio;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.util.ArrayList;
import java.util.List;
/**
 *
 * @author Dr. Carles Torras Font
 */
public class OperacionsBD extends Connexio implements Interface {

    // CONSULTES
    // General de cercar gent
    private static final String CERCA_GENT="SELECT CONCAT (nom, ' ', cognom1, ' ',
    + "cognom2, ' ', nif) as nomt FROM jeroni.professional WHERE nom=? OR
    + "cognom1=? OR cognom2=? UNION ALL SELECT CONCAT(nom, ' ', cognom1, ' ',
    + "cognom2, ' ', nif) as nomt FROM jeroni.alumne WHERE nom=? OR cognom1=?
    + "OR cognom2=?";

    // Professionals -----
    private static final String CREAR_PROFESSIONAL="INSERT INTO jeroni.professional "
    + "(nif,tutoria,lkhorari,nom,cognom1,cognom2, "
    + "password,nomcategoria,actiu) VALUES(?,?,?,?,?,?,?,?)";
    private static final String ELIMINAR_PROFESSIONAL="DELETE FROM jeroni.professional WHERE nif=?";
    private static final String CERCA_PROFESSIONAL="SELECT * FROM jeroni.professional WHERE nif=?";
    private static final String MODIFICAR_PROFESSIONAL="UPDATE jeroni.professional SET tutoria=?,
    + "lkhorari=?,nom=?,cognom1=?,cognom2=?,nomcategoria=?,actiu=? WHERE nif=?";
    private static final String MODIFICAR_FOTOPROFESSIONAL="UPDATE jeroni.professional SET lkfotousuari=? WHERE nif=?";
    private static final String MODIFICAR_PASSWORD="UPDATE jeroni.professional SET password=? WHERE nif=?";
    private static final String LLISTAR_PROF="SELECT * FROM jeroni.professional;";
    private static final String LLISTAR_PROFS="SELECT nom,cognom1,cognom2,nif FROM jeroni.professional;";
    private static final String LLISTAR_TUTORS="SELECT nom,cognom1,cognom2,nif FROM jeroni.professional "
    + "WHERE tutoria='si'";
    private static final String AUTENTICA="SELECT password,nomcategoria,actiu FROM jeroni.professional WHERE nif=?";

    // Alumnes -----
    private static final String CREAR_ALUMNE="INSERT INTO jeroni.alumne (nif,compartit,entrada, "
    + "naixement,nompare,nommare,adreca, "
    + "email,telefon1,telefon2,telefon3,telefon4,diag1,diag2,diag3,diag4,diag5,med1,med2, "
    + "med3,med4,med5,idmun,nom, "
    + "cognom1,cognom2,lkprograma,numtarsan,nomparellapare,nomparellamare,nomtutortutora) "
    + "VALUES(?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?)";
    private static final String ELIMINAR_ALUMNE="DELETE FROM jeroni.alumne WHERE nif=?";
    private static final String MODIFICAR_ALUMNE="UPDATE jeroni.alumne SET compartit=?,entrada=?,
    + "naixement=?,nompare=?,nommare=?,adreca=?,email=?,telefon1=?,telefon2=?,telefon3=?,
    + "telefon4=?,diag1=?,diag2=?,diag3=?,diag4=?,diag5=?,med1=?,med2=?,med3=?,med4=?,
    + "med5=?,idmun=?,nom=?,cognom1=?,cognom2=?,lkprograma=?,numtarsan=?,nomparellapare=?,
    + "nomparellamare=?,nomtutortutora=? WHERE nif=?";
    private static final String CERCA_ALUMNE="SELECT * FROM jeroni.alumne WHERE nif=?";
    private static final String LLISTAR_ALUMNES="SELECT nom,cognom1,cognom2,nif FROM "
    + "jeroni.alumne ORDER BY nom;";
    private static final String LLISTAR_ALUMNE="SELECT * FROM jeroni.alumne;";
    // private static final String ALUMNE_PROF="SELECT nif FROM jeroni.classes WHERE nomaula IN "
    // + "(SELECT nomaula FROM jeroni.aula WHERE professional=?)";
    private static final String ALUMNE_PROF="SELECT nif FROM jeroni.alumclasse JOIN jeroni.profclasse
    + " ON jeroni.alumclasse.nomaula=jeroni.profclasse.elnomaula AND jeroni.alumclasse.curs="
    + "jeroni.profclasse.elcursaula WHERE profclasse.nifprof=? GROUP BY nif;";

    // Alumclasse -----
    private static final String ENTRAR_ALUMNE_CLASSE="INSERT INTO jeroni.alumclasse (nomaula,curs,nif) "
    + "VALUES(?,?,?)";
    private static final String ELIMINAR_ALUMNE_CLASSE="DELETE FROM jeroni.alumclasse WHERE nomaula=? "
    + "AND curs=?";
    private static final String LLISTAR_ALUMNES_CLASSE="SELECT nif FROM jeroni.alumclasse WHERE nomaula=? "
    + "AND curs=?";

    // Profclasse -----
    private static final String ENTRAR_PROF_CLASSE="INSERT INTO jeroni.profclasse (elnomaula,elcursaula,nifprof) "
    + "VALUES(?,?,?)";
    private static final String ELIMINAR_PROF_CLASSE="DELETE FROM jeroni.profclasse WHERE elnomaula=? "
    + "AND elcursaula=?";
    private static final String LLISTAR_PROF_CLASSE="SELECT nifprof FROM jeroni.profclasse WHERE elnomaula=? "
    + "AND elcursaula=?";

    // Classes -----
    private static final String CREAR_CLASSE="INSERT INTO jeroni.classes (nomaula,curs,lkhorari) VALUES(?,?,?)";
    private static final String LLISTAR_CLASSE="SELECT * FROM jeroni.classes;";

```

```

private static final String ELIMINAR_CLASSE="DELETE FROM jeroni.classes WHERE nomaula=? AND curs=?";
private static final String CERCA_CLASSE="SELECT * FROM jeroni.classes WHERE nomaula=? AND curs=?";
private static final String MODIFICAR_CLASSE="UPDATE jeroni.classes SET lkhorari=?
+ "WHERE nomaula=? AND curs=?";
private static final String LLISTAR_CLASSECURS="SELECT nomaula,curs FROM jeroni.classes";

// Aules i cursos -----
private static final String LLISTAR_NOMAULES="SELECT nomaula FROM jeroni.aula";
private static final String ELIMINAR_NOMAULES="DELETE FROM jeroni.aula WHERE nomaula=?";
private static final String CREAR_NOMAULES="INSERT INTO jeroni.aula (nomaula) VALUES(?);";
private static final String LLISTAR_CURSOS="SELECT curs FROM jeroni.cursos";

// Municipis -----
private static final String LLISTAR_MUNICIPIS="SELECT nommunicipi FROM jeroni.municipi";
private static final String CERCA_MUNICIPI="SELECT idmun FROM jeroni.municipi WHERE nommunicipi=?";
private static final String LLISTAR_TOTMUNICIPIS="SELECT * FROM jeroni.municipi";

// Categories -----
private static final String LLISTAR_CATEGORIES="SELECT nomcategoria FROM jeroni.categoria";

// Normes -----
private static final String CREAR_NORMA="INSERT INTO jeroni.normes (titolnorma,lknorma) VALUES(?,?);";
private static final String CERCA_NORMA="SELECT * FROM jeroni.normes WHERE idnormes=?";
private static final String MODIFICAR_NORMA="UPDATE jeroni.normes SET titolnorma=?,lknorma=?
+ "WHERE idnormes=?";
private static final String ELIMINAR_NORMA="DELETE FROM jeroni.normes WHERE idnormes=?";
private static final String LLISTAR_NORMES="SELECT * FROM jeroni.normes";

// Assistencia -----
private static final String CREAR_ASSISTENCIA="INSERT INTO jeroni.assistencia (data,motiu,alumne,
+ "professional) VALUES(?,?,?,?);";
private static final String ELIMINAR_ASSISTENCIA="DELETE FROM jeroni.assistencia WHERE idassistencia=?";
private static final String LLISTAR_ASSISTENCIA="SELECT * FROM jeroni.assistencia WHERE professional LIKE ?";
private static final String LLISTAR_IDASSISTENCIA="SELECT idassistencia FROM jeroni.assistencia "
+ "WHERE professional=?";

// Incidència -----
private static final String CREAR_INCIDENCIA="INSERT INTO jeroni.incidencia (professional,
+ "lkinforme,data,titol) VALUES(?,?,?,?);";
private static final String ELIMINAR_INCIDENCIA="DELETE FROM jeroni.incidencia WHERE idincidencia=?";
private static final String LLISTAR_INCIDENCIA="SELECT * FROM jeroni.incidencia WHERE professional LIKE ?";
private static final String LLISTAR_IDINCIDENCIA="SELECT idincidencia FROM jeroni.incidencia "
+ "WHERE professional=?";
private static final String NUM_INCIDENCIA="SELECT MAX(idincidencia) AS maxim FROM jeroni.incidencia";

// Incialumne -----
private static final String CREAR_INCIALUMNE="INSERT INTO jeroni.incialumne (idincidencia, nif) "
+ "VALUES(?,?);";
private static final String ELIMINAR_INCIALUMNE="DELETE FROM jeroni.incialumne WHERE idincidencia=?";
private static final String LLISTAR_INCIALUMNE="SELECT * FROM jeroni.incialumne WHERE idincidencia IN "
+ "(SELECT idincidencia FROM jeroni.incidencia WHERE professional LIKE ?)";

// Informes -----
private static final String CREAR_INFORME="INSERT INTO jeroni.informe (data, titolinforme,
+ "lkinforme,alumne) VALUES(?,?,?,?);";
private static final String ELIMINAR_INFORME="DELETE FROM jeroni.informe WHERE idinforme=?";
private static final String MODIFICAR_INFORME="UPDATE jeroni.informe SET data=?,
+ "titolinforme=?,lkinforme=?,alumne=? WHERE idinforme=?";
private static final String LLISTAR_INFORME="SELECT * FROM jeroni.informe WHERE "
+ "idinforme IN (SELECT idinforme FROM jeroni.profinformes WHERE nif LIKE ?)";
private static final String LLISTAR_IDINFORME="SELECT idinforme FROM jeroni.informe "
+ "WHERE idinforme IN (SELECT idinforme FROM jeroni.profinformes WHERE nif=?)";
private static final String NUM_INFORME="SELECT MAX(idinforme) AS maxim FROM jeroni.informe";
private static final String CERCA_INFORME="SELECT * FROM jeroni.informe WHERE idinforme=?";

// Profinformes -----
private static final String CREAR_PROFINFORMES="INSERT INTO jeroni.profinformes "
+ "(idinforme, nif) VALUES(?,?);";
private static final String ELIMINAR_PROFINFORMES="DELETE FROM jeroni.profinformes "
+ "WHERE idinforme=?";
private static final String MODIFICAR_PROFINFORMES="UPDATE jeroni.profinformes "
+ "SET nif=? WHERE idinforme=?";
private static final String LLISTAR_PROFINFORMES="SELECT * FROM jeroni.profinformes";
private static final String CERCA_PROFINF="SELECT nif FROM jeroni.profinformes "
+ "WHERE idinforme=?";

// Reunions -----
private static final String CREAR_REUNIO="INSERT INTO jeroni.reunio (datahora, "
+ "assistents,lkcontingut,propera,alumne) VALUES(?,?,?,?,?);";
private static final String ELIMINAR_REUNIO="DELETE FROM jeroni.reunio WHERE idreunio=?";
private static final String MODIFICAR_REUNIO="UPDATE jeroni.reunio SET datahora=?,
+ "assistents=?,lkcontingut=?,propera=?,alumne=? WHERE idreunio=?";
private static final String LLISTAR_REUNIO="SELECT * FROM jeroni.reunio WHERE "
+ "idreunio IN (SELECT idreunio FROM jeroni.profreunions WHERE nif LIKE ?)";
private static final String LLISTAR_IDREUNIO="SELECT idreunio FROM jeroni.reunio "
+ "WHERE idreunio IN (SELECT idreunio FROM jeroni.profreunions WHERE nif=?)";
private static final String NUM_REUNIO="SELECT MAX(idreunio) AS maxim FROM jeroni.reunio";
private static final String CERCA_REUNIO="SELECT * FROM jeroni.reunio WHERE idreunio=?";

// Profreunions -----
private static final String CREAR_PROFREUNIONS="INSERT INTO jeroni.profreunions (nif, "
+ "idreunio) VALUES(?,?);";
private static final String ELIMINAR_PROFREUNIONS="DELETE FROM jeroni.profreunions "
+ "WHERE idreunio=?";
private static final String LLISTAR_PROFREUNIONS="SELECT * FROM jeroni.profreunions";
private static final String CERCA_PROFREUNIONS="SELECT nif FROM jeroni.profreunions "
+ "WHERE idreunio=?";

// Fotografies -----

```



```

private static final String CREAM_FOTO="INSERT INTO jeroni.fotografia (lkfotografia, "
+ "nomaula,curs,etiquetal,etiqueta2,etiqueta3,etiqueta4,datafot) VALUES(?,?,?,?,?,?,?);";
private static final String ELIMINAR_FOTO="DELETE FROM jeroni.fotografia "
+ "WHERE id=?";
private static final String MODIFICAR_FOTO="UPDATE jeroni.fotografia SET lkfotografia=?, "
+ "nomaula=?,curs=?,etiquetal=?,etiqueta2=?,etiqueta3=?,etiqueta4=?,datafot=? WHERE id=?";
private static final String LLISTAR_FOTOS="SELECT * FROM jeroni.fotografia;";
private static final String LLISTAR_IDFOTOS="SELECT id FROM jeroni.fotografia;";
private static final String CERCA_FOTO="SELECT * FROM jeroni.fotografia WHERE id=?";
private static final String LLISTAR_FOTOS_FA="SELECT * FROM jeroni.fotografia WHERE nomaula=? AND curs=?";
private static final String LLISTAR_FOTOS_DA="SELECT * FROM jeroni.fotografia WHERE datafot=? AND datafot<=?";
private static final String LLISTAR_FOTOS_FA_DA="SELECT * FROM jeroni.fotografia WHERE datafot=? AND "
+ "datafot<=? AND nomaula=? AND curs=?";
private static final String LLISTAR_FOTOS_ET1="SELECT * FROM jeroni.fotografia WHERE etiquetal LIKE ? "
+ "OR etiqueta2 LIKE ? OR etiqueta3 LIKE ? OR etiqueta4 LIKE ?";
private static final String LLISTAR_FOTOS_FA_ET1="SELECT * FROM jeroni.fotografia WHERE nomaula=? AND "
+ "curs=? AND etiquetal LIKE ? OR etiqueta2 LIKE ? OR etiqueta3 LIKE ? OR etiqueta4 LIKE ?";
private static final String LLISTAR_FOTOS_DA_ET1="SELECT * FROM jeroni.fotografia WHERE datafot=? AND "
+ "datafot<=? AND etiquetal LIKE ? OR etiqueta2 LIKE ? OR etiqueta3 LIKE ? OR etiqueta4 LIKE ?";
private static final String LLISTAR_FOTOS_FA_DA_ET1="SELECT * FROM jeroni.fotografia WHERE nomaula=? AND "
+ "curs=? AND datafot=? AND datafot<=? AND etiquetal LIKE ? OR etiqueta2 LIKE ? OR etiqueta3 LIKE ? "
+ "OR etiqueta4 LIKE ?";
private static final String LLISTAR_FOTOS_NIF="SELECT * FROM jeroni.fotografia "
+ "WHERE id IN (SELECT id FROM jeroni.fotoalumni WHERE nif=?);";
private static final String LLISTAR_FOTOS_FA_NIF="SELECT * FROM jeroni.fotografia "
+ "WHERE nomaula=? AND curs=? AND id IN (SELECT id FROM jeroni.fotoalumni WHERE nif=?);";
private static final String LLISTAR_FOTOS_DA_NIF="SELECT * FROM jeroni.fotografia "
+ "WHERE datafot=? AND datafot<=? AND id IN (SELECT id FROM jeroni.fotoalumni WHERE nif=?);";
private static final String LLISTAR_FOTOS_FA_DA_NIF="SELECT * FROM jeroni.fotografia "
+ "WHERE datafot=? AND "
+ "datafot<=? AND nomaula=? AND curs=? AND id IN (SELECT id FROM "
+ "jeroni.fotoalumni WHERE nif=?);";
private static final String LLISTAR_FOTOS_ET1_NIF="SELECT * FROM jeroni.fotografia "
+ "WHERE etiquetal LIKE ? "
+ "OR etiqueta2 LIKE ? OR etiqueta3 LIKE ? OR etiqueta4 LIKE ? AND id IN "
+ "(SELECT id FROM jeroni.fotoalumni WHERE nif=?);";
private static final String LLISTAR_FOTOS_FA_ET1_NIF="SELECT * FROM jeroni.fotografia "
+ "WHERE nomaula=? AND "
+ "curs=? AND etiquetal LIKE ? OR etiqueta2 LIKE ? OR etiqueta3 LIKE ? OR "
+ "etiqueta4 LIKE ? AND id IN (SELECT id FROM jeroni.fotoalumni WHERE nif=?);";
private static final String LLISTAR_FOTOS_DA_ET1_NIF="SELECT * FROM jeroni.fotografia "
+ "WHERE datafot=? AND "
+ "datafot<=? AND etiquetal LIKE ? OR etiqueta2 LIKE ? OR etiqueta3 LIKE ? OR "
+ "etiqueta4 LIKE ? AND id IN (SELECT id FROM jeroni.fotoalumni WHERE nif=?);";
private static final String LLISTAR_FOTOS_FA_DA_ET1_NIF="SELECT * FROM jeroni.fotografia WHERE nomaula=? AND "
+ "curs=? AND datafot=? AND datafot<=? AND etiquetal LIKE ? OR etiqueta2 LIKE ? OR etiqueta3 LIKE ? "
+ "OR etiqueta4 LIKE ? AND id IN (SELECT id FROM jeroni.fotoalumni WHERE nif=?);";
private static final String NUM_FOTO="SELECT MAX(id) AS maxim FROM jeroni.fotografia;";

// Fotoalumni -----
private static final String CREAM_FOTOALUMNE="INSERT INTO jeroni.fotoalumni (nif, id) VALUES(?,?);";
private static final String ELIMINAR_FOTOALUMNE="DELETE FROM jeroni.fotoalumni WHERE id=?";
private static final String LLISTAR_FOTOALUMNES="SELECT * FROM jeroni.fotoalumni;";
private static final String LLISTAR_NIFFOTOSALUMNES="SELECT nif FROM jeroni.fotoalumni WHERE id=?";

//Uso del objeto PreparedStatement
private PreparedStatement ps=null;

// Codis t -> codis que indiquen el resultat de l'operaciã:
// Si s'ha creat quelcom correctament, t = 1;
// Si s'ha modificat quelcom correctament, t = 2;
// Si s'ha eliminat quelcom correctament, t = 3;
// Si qualsevol cosa ha fallat, t = 0;

// ----- CERCA GENT -----
@Override
public List<String> cercaGent(String string) {
    List<String> llista = new ArrayList();
    String pel=null;
    abrirConexion();
    try {
        ps=conn.prepareStatement(CERCA_GENT);
        ps.setString(1, string); ps.setString(2, string); ps.setString(3, string);
        ps.setString(4, string); ps.setString(5, string); ps.setString(6, string);
        ResultSet rs =ps.executeQuery();
        while(rs.next()){
            pel = rs.getString("nomt");
            llista.add(pel);
        }
    } catch (SQLException e) {
        e.printStackTrace();
    }
    cerrarConexion();
    return llista;
}

// ----- PROFESSIONALS -----
@Override
public int crearProfessional(Professional professional) {
    abrirConexion ();
    int t=0;
    try {
        conn.setAutoCommit(false);
        ps = conn.prepareStatement(CREAM_PROFESIONAL);
        ps.setString(1, professional.getNif());
        ps.setString(2, professional.getTutoria());
        ps.setString(3, professional.getLkhorari());
        ps.setString(4, professional.getNom());
        ps.setString(5, professional.getCognom1());
    }
}

```

```

        ps.setString(6, professional.getCognom2());
        ps.setString(7, professional.getPassword());
        ps.setString(8, professional.getNomcategoria());
        ps.setString(9, professional.getActiu());
        ps.executeUpdate();
        conn.commit();
        t = 1;
    } catch (SQLException e) {
        try {
            System.out.println("Error, realitzant el rollback: \r\nTipus d'excepció: "+e.toString());
            conn.rollback();
            t = 0;
            System.out.println("Rollback realitzat.");
        } catch (SQLException e1) {
            e1.printStackTrace();
        }
    }
}
cerrarConexion();
return t;
}

@Override
public int modificarProfessional(String nif, String tutor, String arxiu, String nom,
    String cognom1, String cognom2, String categoria, String actiu) {
    abrirConexion ();
    int t=0;
    try {
        conn.setAutoCommit(false);
        ps = conn.prepareStatement(MODIFICAR_PROFESIONAL);
        ps.setString(1, tutor);
        ps.setString(2, arxiu);
        ps.setString(3, nom);
        ps.setString(4, cognom1);
        ps.setString(5, cognom2);
        ps.setString(6, categoria);
        ps.setString(7, actiu);
        ps.setString(8, nif);
        //System.out.println("Operacio: "+ps);
        t = ps.executeUpdate();
        conn.commit();
        t = 2;
    } catch (SQLException e) {
        try {
            System.out.println("Error, realitzant el rollback: \r\nTipus d'excepció: "+e.toString());
            conn.rollback();
            t = 0;
            System.out.println("Rollback realitzat.");
        } catch (SQLException e1) {
            e1.printStackTrace();
        }
    }
}
cerrarConexion();
return t;
}

@Override
public int canviarFotoUsuari(String usuari, String arxiu) {
    abrirConexion ();
    int t=0;
    try {
        conn.setAutoCommit(false);
        ps = conn.prepareStatement(MODIFICAR_FOTOPROFESIONAL);
        ps.setString(1, arxiu);
        ps.setString(2, usuari);
        //System.out.println("Operacio: "+ps);
        t = ps.executeUpdate();
        conn.commit();
    } catch (SQLException e) {
        try {
            System.out.println("Error, realitzant el rollback: \r\nTipus d'excepció: "+e.toString());
            conn.rollback();
            System.out.println("Rollback realitzat.");
        } catch (SQLException e1) {
            e1.printStackTrace();
        }
    }
}
cerrarConexion();
return t;
}

@Override
public int canviPassword(String nif, String clau) {
    abrirConexion ();
    int t=0;
    try {
        conn.setAutoCommit(false);
        ps = conn.prepareStatement(MODIFICAR_PASSWORD);
        ps.setString(1, clau);
        ps.setString(2, nif);
        t = ps.executeUpdate();
        conn.commit();
    } catch (SQLException e) {
        try {
            System.out.println("Error, realitzant el rollback: \r\nTipus d'excepció: "+e.toString());
            conn.rollback();
            System.out.println("Rollback realitzat.");
        } catch (SQLException e1) {
            e1.printStackTrace();
        }
    }
}
}

```

```

    }
    cerrarConexion();
    return t;
}

@Override
public int eliminarProfessional(String nif) {
    abrirConexion();
    int t=0;
    try {
        ps=conn.prepareStatement(ELIMINAR_PROFESSIONAL);
        ps.setString(1, nif);
        //System.out.println(ps);
        ps.executeUpdate();
        t = 3;
    } catch (SQLException e) {
        e.printStackTrace();
        t = 0;
    }
    cerrarConexion();
    return t;
}

@Override
public Professional cercaProfessional(String nif) {
    abrirConexion();
    Professional prof = null;
    try {
        ps=conn.prepareStatement(CERCA_PROFESSIONAL);
        ps.setString(1, nif);
        ResultSet rs =ps.executeQuery();
        while(rs.next()){
            prof=new Professional(null, null, null, null, null, null, null, null, null);
            prof.setNif(rs.getString("nif"));
            prof.setTutoria(rs.getString("tutoria"));
            prof.setLkhorari(rs.getString("lkhorari"));
            prof.setNom(rs.getString("nom"));
            prof.setCognom1(rs.getString("cognom1"));
            prof.setCognom2(rs.getString("cognom2"));
            prof.setPassword(rs.getString("password"));
            prof.setNomcategoria(rs.getString("nomcategoria"));
            prof.setLkfotousuari(rs.getString("lkfotousuari"));
            prof.setActiu(rs.getString("actiu"));
        }
    } catch (SQLException e) {
        e.printStackTrace();
    }
    cerrarConexion();
    return prof;
}

@Override
public List<String> llistarTutors() {
    List<String> llista = new ArrayList();
    String pel=null;
    abrirConexion();
    try {
        ps=conn.prepareStatement(LLISTAR_TUTORS);
        ResultSet rs =ps.executeQuery();
        while(rs.next()){
            pel=rs.getString("nom")+ " "+rs.getString("cognom1")+ " "+rs.getString("cognom2")+", NIF:
"+rs.getString("nif");
            llista.add(pel);
        }
    } catch (SQLException e) {
        e.printStackTrace();
    }
    cerrarConexion();
    return llista;
}

@Override
public List<String> llistarTotsProfs() {
    List<String> llista = new ArrayList();
    String pel=null;
    abrirConexion();
    try {
        ps=conn.prepareStatement(LLISTAR_PROFS);
        ResultSet rs =ps.executeQuery();
        while(rs.next()){
            pel=rs.getString("nom")+ " "+rs.getString("cognom1")+ " "+rs.getString("cognom2")+", NIF:
"+rs.getString("nif");
            llista.add(pel);
        }
    } catch (SQLException e) {
        e.printStackTrace();
    }
    cerrarConexion();
    return llista;
}

@Override
public List<Professional> cercaProf() {
    List<Professional> llista = new ArrayList();
    Professional pel=null;
    abrirConexion();
    try {
        ps=conn.prepareStatement(LLISTAR_PROF);
        ResultSet rs =ps.executeQuery();
        while(rs.next()){

```

```

        pel=new Professional(null, null, null, null, null, null, null, null, null, null);
        pel.setNif(rs.getString("nif"));
        pel.setTutoria(rs.getString("tutoria"));
        pel.setLkhorari(rs.getString("lkhorari"));
        pel.setNom(rs.getString("nom"));
        pel.setCognom1(rs.getString("cognom1"));
        pel.setCognom2(rs.getString("cognom2"));
        pel.setPassword(rs.getString("password"));
        pel.setNomcategoria(rs.getString("nomcategoria"));
        pel.setLkfotousuari(rs.getString("lkfotousuari"));
        pel.setActiu(rs.getString("actiu"));
        llista.add(pel);
    }
} catch (SQLException e) {
    e.printStackTrace();
}
}
cerrarConexion();
return llista;
}

@Override
public List <String> Autenticacio(String usuari) {
    abrirConexion();
    List<String> llista = new ArrayList();
    String resultat;
    try {
        ps=conn.prepareStatement(AUTENTICA);
        ps.setString(1, usuari);
        ResultSet rs =ps.executeQuery();
        while(rs.next()){
            resultat = rs.getString("password");
            llista.add(resultat);
            resultat = rs.getString("nomcategoria");
            llista.add(resultat);
            resultat = rs.getString("actiu");
            llista.add(resultat);
        }
    } catch (SQLException e) {
        e.printStackTrace();
    }
    cerrarConexion();
    return llista;
}

// ----- ALUMNES -----
@Override
public int crearAlumne(Alumne alumne) {
    abrirConexion ();
    int t=0;
    try {
        conn.setAutoCommit(false);
        ps = conn.prepareStatement(CREAR_ALUMNE);
        ps.setString(1, alumne.getNif());
        ps.setString(2, alumne.getCompartit());
        ps.setString(3, alumne.getEntrada());
        ps.setString(4, alumne.getNaixement());
        ps.setString(5, alumne.getNompere());
        ps.setString(6, alumne.getNommare());
        ps.setString(7, alumne.getAdreca());
        ps.setString(8, alumne.getEmail());
        ps.setString(9, alumne.getTelefon1());
        ps.setString(10, alumne.getTelefon2());
        ps.setString(11, alumne.getTelefon3());
        ps.setString(12, alumne.getTelefon4());
        ps.setString(13, alumne.getDiag1());
        ps.setString(14, alumne.getDiag2());
        ps.setString(15, alumne.getDiag3());
        ps.setString(16, alumne.getDiag4());
        ps.setString(17, alumne.getDiag5());
        ps.setString(18, alumne.getMed1());
        ps.setString(19, alumne.getMed2());
        ps.setString(20, alumne.getMed3());
        ps.setString(21, alumne.getMed4());
        ps.setString(22, alumne.getMed5());
        ps.setInt(23, alumne.getIdmun());
        ps.setString(24, alumne.getNom());
        ps.setString(25, alumne.getCognom1());
        ps.setString(26, alumne.getCognom2());
        ps.setString(27, alumne.getPrograma());
        ps.setString(28, alumne.getNumtarsan());
        ps.setString(29, alumne.getNomparellapare());
        ps.setString(30, alumne.getNomparellamare());
        ps.setString(31, alumne.getNomtutortutora());
        ps.executeUpdate();
        conn.commit();
        t = 1;
    } catch (SQLException e) {
        try {
            System.out.println("Error, realitzant el rollback: \r\nTipus d'excepció: "+e.toString());
            conn.rollback();
            System.out.println("Rollback realitzat.");
            t = 0;
        } catch (SQLException e1) {
            e1.printStackTrace();
        }
    }
}
cerrarConexion();
return t;
}

```

```

@Override
public int modificarAlumne(Alumne alumne) {
    abrirConexion ();
    int t=0;
    try {
        conn.setAutoCommit(false);
        ps = conn.prepareStatement(MODIFICAR_ALUMNE);
        ps.setString(1, alumne.getCompartit());
        ps.setString(2, alumne.getEntrada());
        ps.setString(3, alumne.getNaixement());
        ps.setString(4, alumne.getNompere());
        ps.setString(5, alumne.getNommare());
        ps.setString(6, alumne.getAdreca());
        ps.setString(7, alumne.getEmail());
        ps.setString(8, alumne.getTelefon1());
        ps.setString(9, alumne.getTelefon2());
        ps.setString(10, alumne.getTelefon3());
        ps.setString(11, alumne.getTelefon4());
        ps.setString(12, alumne.getDiag1());
        ps.setString(13, alumne.getDiag2());
        ps.setString(14, alumne.getDiag3());
        ps.setString(15, alumne.getDiag4());
        ps.setString(16, alumne.getDiag5());
        ps.setString(17, alumne.getMed1());
        ps.setString(18, alumne.getMed2());
        ps.setString(19, alumne.getMed3());
        ps.setString(20, alumne.getMed4());
        ps.setString(21, alumne.getMed5());
        ps.setInt(22, alumne.getIdmun());
        ps.setString(23, alumne.getNom());
        ps.setString(24, alumne.getCognom1());
        ps.setString(25, alumne.getCognom2());
        ps.setString(26, alumne.getPrograma());
        ps.setString(27, alumne.getNumtarsan());
        ps.setString(28, alumne.getNomparellapare());
        ps.setString(29, alumne.getNomparellamare());
        ps.setString(30, alumne.getNomtutortutora());
        ps.setString(31, alumne.getNif());
        //System.out.println("Operacio: "+ps);
        ps.executeUpdate();
        conn.commit();
        t = 2;
    } catch (SQLException e) {
        try {
            System.out.println("Error, realitzant el rollback: \r\nTipus d'excepció: "+e.toString());
            conn.rollback();
            System.out.println("Rollback realitzat.");
            t = 0;
        } catch (SQLException e1) {
            e1.printStackTrace();
        }
    }
    cerrarConexion();
    return t;
}

@Override
public Alumne cercaAlumne(String nif) {
    abrirConexion();
    Alumne alumne = null;
    try {
        ps=conn.prepareStatement(CERCA_ALUMNE);
        ps.setString(1, nif);
        ResultSet rs =ps.executeQuery();
        while(rs.next()){
            alumne = new Alumne(null, null, null, null, null, null, null, null, null,
                null, null, null, null, null, null, null, null, null, null,
                null, null, 0, null, null, null, null, null, null, null, null);
            alumne.setNif(rs.getString("nif"));
            alumne.setCompartit(rs.getString("compartit"));
            alumne.setEntrada(rs.getString("entrada"));
            alumne.setNaixement(rs.getString("naixement"));
            alumne.setNompere(rs.getString("nompere"));
            alumne.setNommare(rs.getString("nommare"));
            alumne.setAdreca(rs.getString("adreca"));
            alumne.setEmail(rs.getString("email"));
            alumne.setTelefon1(rs.getString("telefon1"));
            alumne.setTelefon2(rs.getString("telefon2"));
            alumne.setTelefon3(rs.getString("telefon3"));
            alumne.setTelefon4(rs.getString("telefon4"));
            alumne.setDiag1(rs.getString("diag1"));
            alumne.setDiag2(rs.getString("diag2"));
            alumne.setDiag3(rs.getString("diag3"));
            alumne.setDiag4(rs.getString("diag4"));
            alumne.setDiag5(rs.getString("diag5"));
            alumne.setMed1(rs.getString("med1"));
            alumne.setMed2(rs.getString("med2"));
            alumne.setMed3(rs.getString("med3"));
            alumne.setMed4(rs.getString("med4"));
            alumne.setMed5(rs.getString("med5"));
            alumne.setIdmun(rs.getInt("idmun"));
            alumne.setNom(rs.getString("nom"));
            alumne.setCognom1(rs.getString("cognom1"));
            alumne.setCognom2(rs.getString("cognom2"));
            alumne.setPrograma(rs.getString("lkprograma"));
            alumne.setNumtarsan(rs.getString("numtarsan"));
            alumne.setNomparellapare(rs.getString("nomparellapare"));
            alumne.setNomparellamare(rs.getString("nomparellamare"));
            alumne.setNomtutortutora(rs.getString("nomtutortutora"));
        }
    }
}

```

```

    }
    } catch (SQLException e) {
        e.printStackTrace();
    }
    cerrarConexion();
    return alumne;
}

@Override
public List<String> llistarTotsAlumnes() {
    List<String> llista = new ArrayList();
    String pel=null;
    obrirConexion();
    try {
        ps=conn.prepareStatement(LLISTAR_ALUMNES);
        ResultSet rs =ps.executeQuery();
        while(rs.next()){
            pel=rs.getString("nom")+ " "+rs.getString("cognom1")+ " "+rs.getString("cognom2")+", NIF:
"+rs.getString("nif");
            llista.add(pel);
        }
    } catch (SQLException e) {
        e.printStackTrace();
    }
    cerrarConexion();
    return llista;
}

@Override
public List <Alumne> cercaAlumne() {
    List<Alumne> llista = new ArrayList();
    Alumne pel=null;
    obrirConexion();
    try {
        ps=conn.prepareStatement(LLISTAR_ALUMNE);
        ResultSet rs =ps.executeQuery();
        while(rs.next()){
            pel=new Alumne(null, null, null, null, null, null, null, null,
                null, null, null, null, null, null, null, null,
                null, null, null, null, null, 0, null, null, null, null, null, null, null);
            pel.setNif(rs.getString("nif"));
            pel.setCompartit(rs.getString("compartit"));
            pel.setEntrada(rs.getString("entrada"));
            pel.setNaixement(rs.getString("naixement"));
            pel.setNompere(rs.getString("nompere"));
            pel.setNompare(rs.getString("nompare"));
            pel.setAdreca(rs.getString("adreca"));
            pel.setEmail(rs.getString("email"));
            pel.setTelefon1(rs.getString("telefon1"));
            pel.setTelefon2(rs.getString("telefon2"));
            pel.setTelefon3(rs.getString("telefon3"));
            pel.setTelefon4(rs.getString("telefon4"));
            pel.setDiag1(rs.getString("diag1"));
            pel.setDiag2(rs.getString("diag2"));
            pel.setDiag3(rs.getString("diag3"));
            pel.setDiag4(rs.getString("diag4"));
            pel.setDiag5(rs.getString("diag5"));
            pel.setMed1(rs.getString("med1"));
            pel.setMed2(rs.getString("med2"));
            pel.setMed3(rs.getString("med3"));
            pel.setMed4(rs.getString("med4"));
            pel.setMed5(rs.getString("med5"));
            pel.setIdmun(rs.getInt("idmun"));
            pel.setNom(rs.getString("nom"));
            pel.setCognom1(rs.getString("cognom1"));
            pel.setCognom2(rs.getString("cognom2"));
            pel.setPrograma(rs.getString("lkprograma"));
            pel.setNumtarsan(rs.getString("numtarsan"));
            pel.setNompereIlapare(rs.getString("nompereIlapare"));
            pel.setNompereIllamare(rs.getString("nompereIllamare"));
            pel.setNomtutorIutora(rs.getString("nomtutorIutora"));
            llista.add(pel);
        }
    } catch (SQLException e) {
        e.printStackTrace();
    }
    cerrarConexion();
    return llista;
}

@Override
public int eliminarAlumne(String nif) {
    obrirConexion();
    int t=0;
    try {
        ps=conn.prepareStatement(ELIMINAR_ALUMNE);
        ps.setString(1, nif);
        ps.executeUpdate();
        t = 3;
    } catch (SQLException e) {
        e.printStackTrace();
        t = 0;
    }
    cerrarConexion();
    return t;
}

@Override
public List <String> NifAlumnesProf(String nif) {
    List<String> llista = new ArrayList();

```

```

String pel = null;
abrirConexion();
try {
    ps=conn.prepareStatement(ALUMNE_PROF);
    ps.setString(1, nif);
    ResultSet rs =ps.executeQuery();
    //System.out.println(ps);
    while(rs.next()){
        pel = rs.getString("nif");
        llista.add(pel);
    }
} catch (SQLException e) {
    e.printStackTrace();
}
cerrarConexion();
return llista;
}

// ----- ALUMCLASSES -----
@Override
public int insertarAlumesClasse(String nomaula, String curs, List<String> nif) {
    abrirConexion();
    int t=0;
    try {
        ps=conn.prepareStatement(ELIMINAR_ALUMNE_CLASSE);
        ps.setString(1, nomaula);
        ps.setString(2, curs);
        ps.executeUpdate();
        //System.out.println("Hola "+ps);
        ps=conn.prepareStatement(ENTRAR_ALUMNE_CLASSE);
        for (int i = 0; i < nif.size(); i++) {
            String lonif = nif.get(i);
            ps.setString(1, nomaula);
            ps.setString(2, curs);
            ps.setString(3, lonif);
            ps.addBatch();
            if ((i + 1) % 1000 == 0) {
                //System.out.println("Adeul "+ps);
                ps.executeBatch(); // Execute every 1000 items.
            }
        }
        //System.out.println("Adeu2 "+ps);
        ps.executeBatch();
        t = 1;
    } catch (SQLException e) {
        e.printStackTrace();
        t = 0;
    }
    cerrarConexion();
    return t;
}

@Override
public int eliminarAlumesClasse(String nomaula, String curs) {
    abrirConexion();
    int t=0;
    try {
        ps=conn.prepareStatement(ELIMINAR_ALUMNE_CLASSE);
        ps.setString(1, nomaula);
        ps.setString(2, curs);
        ps.executeUpdate();
        t = 3;
    } catch (SQLException e) {
        e.printStackTrace();
        t = 0;
    }
    cerrarConexion();
    return t;
}

@Override
public List<String> llistarAlumesClasse(String nomaula, String curs) {
    List<String> llista = new ArrayList();
    abrirConexion();
    String nif = null;
    try {
        ps=conn.prepareStatement(LLISTAR_ALUMNES_CLASSE);
        ps.setString(1, nomaula);
        ps.setString(2, curs);
        ResultSet rs =ps.executeQuery();
        while(rs.next()){
            nif = rs.getString("nif");
            llista.add(nif);
        }
    } catch (SQLException e) {
        e.printStackTrace();
    }
    cerrarConexion();
    return llista;
}

// ----- PROFCLASSES -----
@Override
public int insertarProfesClasse(String elnomaula, String elcursa, List<String> nifprof) {
    abrirConexion();
    int t=0;
    try {
        ps=conn.prepareStatement(ELIMINAR_PROF_CLASSE);
        ps.setString(1, elnomaula);

```

```

        ps.setString(2, elcursaula);
        ps.executeUpdate();
        //System.out.println("Elimino "+ps);
        ps=conn.prepareStatement(ENTRAR_PROF_CLASSE);
        for (int i = 0; i < nifprof.size(); i++) {
            String nif = nifprof.get(i);
            ps.setString(1, elnomaula);
            ps.setString(2, elcursaula);
            ps.setString(3, nif);
            ps.addBatch();
            if ((i + 1) % 1000 == 0) {
                //System.out.println("Entro: "+ps);
                ps.executeBatch(); // Execute every 1000 items.
            }
        }
        //System.out.println("Entro: "+ps);
        ps.executeBatch();
        t = 1;
    } catch (SQLException e) {
        e.printStackTrace();
        t = 0;
    }
}
cerrarConexion();
return t;
}

@Override
public int eliminarProfesClasse(String elnomaula, String elcursaula) {
    abrirConexion();
    int t=0;
    try {
        ps=conn.prepareStatement(ELIMINAR_PROF_CLASSE);
        ps.setString(1, elnomaula);
        ps.setString(2, elcursaula);
        ps.executeUpdate();
        t = 3;
    } catch (SQLException e) {
        e.printStackTrace();
        t = 0;
    }
    cerrarConexion();
    return t;
}

@Override
public List<String> llistarProfesClasse(String elnomaula, String elcursaula) {
    List<String> llista = new ArrayList();
    abrirConexion();
    String nif = null;
    try {
        ps=conn.prepareStatement(LLISTAR_PROF_CLASSE);
        ps.setString(1, elnomaula);
        ps.setString(2, elcursaula);
        ResultSet rs =ps.executeQuery();
        while(rs.next()){
            nif = rs.getString("nifprof");
            llista.add(nif);
        }
    } catch (SQLException e) {
        e.printStackTrace();
    }
    cerrarConexion();
    return llista;
}

// ----- CLASSES -----
@Override
public int crearClasse(Clases aula) {
    abrirConexion ();
    int t=0;
    try {
        conn.setAutoCommit(false);
        ps = conn.prepareStatement(CREAR_CLASSE);
        ps.setString(1, aula.getNomaula());
        ps.setString(2, aula.getCurs());
        ps.setString(3, aula.getLkhorari());
        ps.executeUpdate();
        conn.commit();
        t = 1;
    } catch (SQLException e) {
        try {
            System.out.println("Error, realitzant el rollback: \r\nTipus d'excepció: "+e.toString());
            conn.rollback();
            System.out.println("Rollback realitzat.");
            t = 0;
        } catch (SQLException e1) {
            e1.printStackTrace();
        }
    }
    cerrarConexion();
    return t;
}

@Override
public List<Clases> llistarClasses() {
    List<Clases> llista = new ArrayList();
    abrirConexion();
    Clases pel = null;
    try {

```



```

        ps=conn.prepareStatement(LLISTAR_CLASSE);
        ResultSet rs =ps.executeQuery();
        while(rs.next()){
            pel=new Classes(null, null, null);
            pel.setNomaula(rs.getString("nomaula"));
            pel.setCurs(rs.getString("curs"));
            pel.setLkhorari(rs.getString("lkhorari"));
            llista.add(pel);
        }
    } catch (SQLException e) {
        e.printStackTrace();
    }
    cerrarConexion();
    return llista;
}

@Override
public List<String> llistarClasseCurs() {
    List<String> llista = new ArrayList();
    obrirConexion();
    String pel = null;
    try {
        ps=conn.prepareStatement(LLISTAR_CLASSECURS);
        ResultSet rs =ps.executeQuery();
        while(rs.next()){
            pel = rs.getString("nomaula") + ", " + rs.getString("curs");
            llista.add(pel);
        }
    } catch (SQLException e) {
        e.printStackTrace();
    }
    cerrarConexion();
    return llista;
}

@Override
public int eliminarClasse(String nom, String curs) {
    obrirConexion();
    int t=0;
    try {
        ps=conn.prepareStatement(ELIMINAR_CLASSE);
        ps.setString(1, nom);
        ps.setString(2, curs);
        //System.out.println(ps);
        ps.executeUpdate();
        t = 3;
    } catch (SQLException e) {
        e.printStackTrace();
        t = 0;
    }
    cerrarConexion();
    return t;
}

@Override
public Classes cercaClasse(String nomaula, String curs) {
    obrirConexion();
    Classes aula = null;
    try {
        ps=conn.prepareStatement(CERCA_CLASSE);
        ps.setString(1, nomaula);
        ps.setString(2, curs);
        ResultSet rs =ps.executeQuery();
        while(rs.next()){
            aula=new Classes(null, null, null);
            aula.setNomaula(rs.getString("nomaula"));
            aula.setCurs(rs.getString("curs"));
            aula.setLkhorari(rs.getString("lkhorari"));
        }
    } catch (SQLException e) {
        e.printStackTrace();
    }
    cerrarConexion();
    return aula;
}

@Override
public int modificarClasse(Classes classe) {
    obrirConexion ();
    int t=0;
    try {
        conn.setAutoCommit(false);
        ps = conn.prepareStatement(MODIFICAR_CLASSE);
        ps.setString(1, classe.getLkhorari());
        ps.setString(2, classe.getNomaula());
        ps.setString(3, classe.getCurs());
        //System.out.println("Operacio: "+ps);
        ps.executeUpdate();
        conn.commit();
        t = 2;
    } catch (SQLException e) {
        try {
            System.out.println("Error, realitzant el rollback: \r\nTipus d'exceptiÃ³: "+e.toString());
            conn.rollback();
            System.out.println("Rollback realitzat.");
            t = 0;
        } catch (SQLException e1) {
            e1.printStackTrace();
            t = 0;
        }
    }
}

```

```

    }
    cerrarConexion();
    return t;
}

// ----- AULES i CURSOS -----
@Override
public List<String> llistarNomAules() {
    List<String> llista = new ArrayList();
    String pel=null;
    abrirConexion();
    try {
        ps=conn.prepareStatement(LLISTAR_NOMAULES);
        ResultSet rs =ps.executeQuery();
        while(rs.next()){
            pel=rs.getString("nomaula");
            llista.add(pel);
        }
    } catch (SQLException e) {
        e.printStackTrace();
    }
    cerrarConexion();
    return llista;
}

@Override
public int crearNomAules(Aula aules) {
    abrirConexion ();
    int t=0;
    try {
        conn.setAutoCommit(false);
        ps = conn.prepareStatement(CREAR_NOMAULES);
        ps.setString(1, aules.getNomaula());
        ps.executeUpdate();
        conn.commit();
        t = 1;
    } catch (SQLException e) {
        try {
            System.out.println("Error, realitzant el rollback: \r\nTipus d'excepciÃ³: "+e.toString());
            conn.rollback();
            System.out.println("Rollback realitzat.");
            t = 0;
        } catch (SQLException e1) {
            e1.printStackTrace();
            t = 0;
        }
    }
    cerrarConexion();
    return t;
}

@Override
public int eliminarNomAules(String nom) {
    abrirConexion();
    int t=0;
    try {
        ps=conn.prepareStatement(ELIMINAR_NOMAULES);
        ps.setString(1, nom);
        ps.executeUpdate();
        t = 3;
    } catch (SQLException e) {
        e.printStackTrace();
        t = 0;
    }
    cerrarConexion();
    return t;
}

@Override
public List<String> llistarCursos() {
    List<String> llista = new ArrayList();
    abrirConexion();
    String curs = null;
    try {
        ps=conn.prepareStatement(LLISTAR_CURSOS);
        ResultSet rs =ps.executeQuery();
        while(rs.next()){
            curs = rs.getString("curs");
            llista.add(curs);
        }
    } catch (SQLException e) {
        e.printStackTrace();
    }
    cerrarConexion();
    return llista;
}

// ----- MUNICIPIIS -----
@Override
public List<String> llistarMunicipis() {
    List<String> llista = new ArrayList();
    String pel=null;
    abrirConexion();
    try {
        ps=conn.prepareStatement(LLISTAR_MUNICIPIS);
        ResultSet rs =ps.executeQuery();
        while(rs.next()){
            pel=rs.getString("nommunicipi");

```

```

        llista.add(pel);
    }
} catch (SQLException e) {
    e.printStackTrace();
}
}
cerrarConexion();
return llista;
}

@Override
public List<Municipi> llistarTotMunicipis() {
    List<Municipi> llista = new ArrayList();
    obrirConexion();
    Municipi pel = null;
    try {
        ps=conn.prepareStatement(LLISTAR_TOTMUNICIPIS);
        ResultSet rs =ps.executeQuery();
        while(rs.next()){
            pel=new Municipi(0, null, null);
            pel.setIdmun(rs.getInt("idmun"));
            pel.setNommunicipi(rs.getString("nommunicipi"));
            pel.setNomprovincia(rs.getString("nomprovincia"));
            llista.add(pel);
        }
    } catch (SQLException e) {
        e.printStackTrace();
    }
    cerrarConexion();
    return llista;
}

@Override
public int cercarMunicipi(String nom) {
    int resposta = 0;
    obrirConexion();
    try {
        ps=conn.prepareStatement(CERCA_MUNICIPI);
        ps.setString(1, nom);
        ResultSet rs =ps.executeQuery();
        while(rs.next()){
            resposta = rs.getInt("idmun");
        }
    } catch (SQLException e) {
        e.printStackTrace();
    }
    cerrarConexion();
    return resposta;
}

// ----- CATEGORIES -----
@Override
public List<String> llistarCategories() {
    List<String> llista = new ArrayList();
    String pel=null;
    obrirConexion();
    try {
        ps=conn.prepareStatement(LLISTAR_CATEGORIES);
        ResultSet rs =ps.executeQuery();
        while(rs.next()){
            pel=rs.getString("nomcategoria");
            llista.add(pel);
        }
    } catch (SQLException e) {
        e.printStackTrace();
    }
    cerrarConexion();
    return llista;
}

// ----- NORMES -----
@Override
public int crearNorma(String titol, String arxiu) {
    obrirConexion ();
    int t=0;
    try {
        conn.setAutoCommit(false);
        ps = conn.prepareStatement(CREAR_NORMA);
        ps.setString(1, titol);
        ps.setString(2, arxiu);
        ps.executeUpdate();
        t = 1;
        conn.commit();
    } catch (SQLException e) {
        try {
            System.out.println("Error, realitzant el rollback: \r\nTipus d'excepciÃ³: "+e.toString());
            conn.rollback();
            System.out.println("Rollback realitzat.");
            t = 0;
        } catch (SQLException e1) {
            e1.printStackTrace();
            t = 0;
        }
    }
    cerrarConexion();
    return t;
}

@Override
public Normes cercaNorma(String id) {

```

```

    obrirConexion();
    Normes norma = null;
    try {
        ps=conn.prepareStatement(CERCA_NORMA);
        ps.setString(1, id);
        ResultSet rs =ps.executeQuery();
        while(rs.next()){
            norma = new Normes(null, null, null);
            norma.setIdnormes(rs.getString("idnormes"));
            norma.setTitolnorma(rs.getString("titolnorma"));
            norma.setLknorma(rs.getString("lknorma"));
        }
    } catch (SQLException e) {
        e.printStackTrace();
    }
    cerrarConexion();
    return norma;
}

@Override
public int modificarNorma(Normes norma) {
    obrirConexion ();
    int t=0;
    try {
        conn.setAutoCommit(false);
        ps = conn.prepareStatement(MODIFICAR_NORMA);
        ps.setString(1, norma.getTitolnorma());
        ps.setString(2, norma.getLknorma());
        ps.setString(3, norma.getIdnormes());
        //System.out.println("Operacio: "+ps);
        ps.executeUpdate();
        conn.commit();
        t = 2;
    } catch (SQLException e) {
        try {
            System.out.println("Error, realitzant el rollback: \r\nTipus d'excepció: "+e.toString());
            conn.rollback();
            System.out.println("Rollback realitzat.");
            t = 0;
        } catch (SQLException e1) {
            e1.printStackTrace();
            t = 0;
        }
    }
    cerrarConexion();
    return t;
}

@Override
public int eliminarNorma(String id) {
    obrirConexion();
    int t=0;
    try {
        ps=conn.prepareStatement(ELIMINAR_NORMA);
        ps.setString(1, id);
        ps.executeUpdate();
        t = 3;
        //System.out.println(ps);
    } catch (SQLException e) {
        e.printStackTrace();
        t = 0;
    }
    cerrarConexion();
    return t;
}

@Override
public List<Normes> llistarNormes() {
    List<Normes> llista = new ArrayList();
    obrirConexion();
    Normes pel = null;
    try {
        ps=conn.prepareStatement(LLISTAR_NORMES);
        ResultSet rs =ps.executeQuery();
        while(rs.next()){
            pel=new Normes(null, null, null);
            pel.setIdnormes(rs.getString("idnormes"));
            pel.setTitolnorma(rs.getString("titolnorma"));
            pel.setLknorma(rs.getString("lknorma"));
            llista.add(pel);
        }
    } catch (SQLException e) {
        e.printStackTrace();
    }
    cerrarConexion();
    return llista;
}

@Override
public List<String> llistarIdNormes() {
    List<String> llista = new ArrayList();
    obrirConexion();
    String pel = null;
    try {
        ps=conn.prepareStatement(LLISTAR_NORMES);
        ResultSet rs =ps.executeQuery();
        while(rs.next()){
            pel=rs.getString("idnormes");
            llista.add(pel);
        }
    }
}

```

```

    } catch (SQLException e) {
        e.printStackTrace();
    }
    cerrarConexion();
    return llista;
}

// ----- ASSISTENCIA -----
@Override
public int crearAssistencia(String data, String motiu, String alumne, String professional) {
    abrirConexion ();
    int t=0;
    try {
        conn.setAutoCommit(false);
        ps = conn.prepareStatement(CREAR_ASSISTENCIA);
        ps.setString(1, data);
        ps.setString(2, motiu);
        ps.setString(3, alumne);
        ps.setString(4, professional);
        ps.executeUpdate();
        conn.commit();
        t = 1;
    } catch (SQLException e) {
        try {
            System.out.println("Error, realitzant el rollback: \r\nTipus d'excepció: "+e.toString());
            conn.rollback();
            System.out.println("Rollback realitzat.");
            t = 0;
        } catch (SQLException e1) {
            e1.printStackTrace();
            t = 0;
        }
    }
    cerrarConexion();
    return t;
}

@Override
public int eliminarAssistencia(String id) {
    abrirConexion();
    int t=0;
    try {
        ps=conn.prepareStatement(ELIMINAR_ASSISTENCIA);
        ps.setString(1, id);
        ps.executeUpdate();
        t = 3;
        //System.out.println(ps);
    } catch (SQLException e) {
        e.printStackTrace();
        t = 0;
    }
    cerrarConexion();
    return t;
}

@Override
public List<Assistencia> llistarAssistencia(String nif) {
    List<Assistencia> llista = new ArrayList();
    Assistencia pel=null;
    abrirConexion();
    try {
        ps=conn.prepareStatement(LLISTAR_ASSISTENCIA);
        ps.setString(1, nif);
        //System.out.println(ps);
        ResultSet rs =ps.executeQuery();
        while(rs.next()){
            pel=new Assistencia(0, null, null, null, null);
            pel.setIdassistencia(rs.getInt("idassistencia"));
            pel.setData(rs.getString("data"));
            pel.setMotiu(rs.getString("motiu"));
            pel.setAlumne(rs.getString("alumne"));
            pel.setProfessional(rs.getString("professional"));
            llista.add(pel);
        }
    } catch (SQLException e) {
        e.printStackTrace();
    }
    cerrarConexion();
    return llista;
}

@Override
public List<String> llistarIdAssistencia(String nif) {
    List<String> llista = new ArrayList();
    String pel=null;
    abrirConexion();
    try {
        ps=conn.prepareStatement(LLISTAR_IDASSISTENCIA);
        ps.setString(1, nif);
        //System.out.println(ps);
        ResultSet rs =ps.executeQuery();
        while(rs.next()){
            pel=rs.getString("idassistencia");
            llista.add(pel);
        }
    } catch (SQLException e) {
        e.printStackTrace();
    }
    cerrarConexion();
}

```

```

        return llista;
    }
}
// ----- INCIDÈNCIA -----
@Override
public int crearIncidencia(String professional, String lkinforme, String data, String titol) {
    abrirConexion ();
    int t=0;
    try {
        // Creo la incidència
        conn.setAutoCommit(false);
        ps = conn.prepareStatement(CREAR_INCIDENCIA);
        ps.setString(1, professional);
        ps.setString(2, lkinforme);
        ps.setString(3, data);
        ps.setString(4, titol);
        ps.executeUpdate();
        conn.commit();
        // Pillo el nÃºmero de incidència que he creat
        conn.setAutoCommit(false);
        ps = conn.prepareStatement(NUM_INCIDENCIA);
        ResultSet rs =ps.executeQuery();
        while(rs.next()){
            t = Integer.parseInt(rs.getString("maxim"));
        }
        conn.commit();
    } catch (SQLException e) {
        try {
            System.out.println("Error, realitzant el rollback: \r\nTipus d'excepció: "+e.toString());
            conn.rollback();
            System.out.println("Rollback realitzat.");
        } catch (SQLException e1) {
            e1.printStackTrace();
        }
    }
    cerrarConexion();
    return t;
}

@Override
public int eliminarIncidencia(String id) {
    abrirConexion();
    int t=0;
    try {
        ps=conn.prepareStatement(ELIMINAR_INCIDENCIA);
        ps.setString(1, id);
        ps.executeUpdate();
        t = 3;
    } catch (SQLException e) {
        e.printStackTrace();
        t = 0;
    }
    cerrarConexion();
    return t;
}

@Override
public List<Incidencia> llistarIncidencia(String nif) {
    List<Incidencia> llista = new ArrayList();
    Incidencia pel=null;
    abrirConexion();
    try {
        ps=conn.prepareStatement(LLISTAR_INCIDENCIA);
        ps.setString(1, nif);
        ResultSet rs =ps.executeQuery();
        //System.out.println(ps);
        while(rs.next()){
            pel=new Incidencia(0, null, null, null, null);
            pel.setIdincidencia(rs.getInt("idincidencia"));
            pel.setNif(rs.getString("professional"));
            pel.setLkinforme(rs.getString("lkinforme"));
            pel.setData(rs.getString("data"));
            pel.setTitol(rs.getString("titol"));
            llista.add(pel);
        }
    } catch (SQLException e) {
        e.printStackTrace();
    }
    cerrarConexion();
    return llista;
}

@Override
public List<String> llistarIdIncidencia(String nif) {
    List<String> llista = new ArrayList();
    String pel=null;
    abrirConexion();
    try {
        ps=conn.prepareStatement(LLISTAR_IDINCIDENCIA);
        ps.setString(1, nif);
        //System.out.println(ps);
        ResultSet rs =ps.executeQuery();
        while(rs.next()){
            pel=rs.getString("idincidencia");
            llista.add(pel);
        }
    } catch (SQLException e) {
        e.printStackTrace();
    }
    cerrarConexion();
}

```

```

    }
    return llista;
}

// ----- INCIALUMNES -----
@Override
public int crearIncialumne(String numinci, List<String> alumnes) {
    abrirConexion ();
    int t=0;
    try {
        conn.setAutoCommit(false);
        ps = conn.prepareStatement(CREAR_INCIALUMNE);
        for (int i = 0; i < alumnes.size(); i++) {
            ps.setString(1, numinci);
            ps.setString(2, alumnes.get(i));
            ps.addBatch();
        }
        ps.executeBatch();
        conn.commit();
        t = 1;
    } catch (SQLException e) {
        try {
            System.out.println("Error, realitzant el rollback: \r\nTipus d'excepció: "+e.toString());
            conn.rollback();
            System.out.println("Rollback realitzat.");
            t = 0;
        } catch (SQLException e1) {
            e1.printStackTrace();
            t = 0;
        }
    }
    cerrarConexion();
    return t;
}

@Override
public int eliminarIncialumne(String id) {
    abrirConexion();
    int t=0;
    try {
        ps=conn.prepareStatement(ELIMINAR_INCIALUMNE);
        ps.setString(1, id);
        ps.executeUpdate();
        t = 3;
    } catch (SQLException e) {
        e.printStackTrace();
        t = 0;
    }
    cerrarConexion();
    return t;
}

@Override
public List <Incialumne> llistarIncialumne(String nif) {
    List<Incialumne> llista = new ArrayList();
    Incialumne pel=null;
    abrirConexion();
    try {
        ps=conn.prepareStatement(LLISTAR_INCIALUMNE);
        ps.setString(1, nif);
        ResultSet rs =ps.executeQuery();
        while(rs.next()){
            pel=new Incialumne(0, null);
            pel.setIdincidencia(rs.getInt("idincidencia"));
            pel.setNif(rs.getString("nif"));
            llista.add(pel);
        }
    } catch (SQLException e) {
        e.printStackTrace();
    }
    cerrarConexion();
    return llista;
}

// ----- INFORMES -----
@Override
public int crearInforme(String alumne, String lkinforme, String data, String titol) {
    abrirConexion ();
    int t=0;
    try {
        // Creo la incidència
        conn.setAutoCommit(false);
        ps = conn.prepareStatement(CREAR_INFORME);
        ps.setString(1, data);
        ps.setString(2, titol);
        ps.setString(3, lkinforme);
        ps.setString(4, alumne);
        t = ps.executeUpdate();
        conn.commit();
        // Pillo el nÃºmero de incidència que he creat
        conn.setAutoCommit(false);
        ps = conn.prepareStatement(NUM_INFORME);
        ResultSet rs =ps.executeQuery();
        while(rs.next()){
            t=Integer.parseInt(rs.getString("maxim"));
        }
        conn.commit();
    } catch (SQLException e) {
        try {
            System.out.println("Error, realitzant el rollback: \r\nTipus d'excepció: "+e.toString());

```

```

        conn.rollback();
        System.out.println("Rollback realitzat.");
        t = 0;
    } catch (SQLException e1) {
        e1.printStackTrace();
        t = 0;
    }
}
cerrarConexion();
return t;
}

@Override
public int eliminarInforme(String id) {
    abrirConexion();
    int t=0;
    try {
        ps=conn.prepareStatement(ELIMINAR_INFORME);
        ps.setString(1, id);
        ps.executeUpdate();
        t = 3;
    } catch (SQLException e) {
        e.printStackTrace();
        t = 0;
    }
    cerrarConexion();
    return t;
}

@Override
public List <Informe> llistarInformes(String nif) {
    List<Informe> llista = new ArrayList();
    Informe pel=null;
    abrirConexion();
    try {
        ps=conn.prepareStatement(LLISTAR_INFORME);
        ps.setString(1, nif);
        ResultSet rs =ps.executeQuery();
        while(rs.next()){
            pel=new Informe(0, null, null, null, null);
            pel.setIdinforme(rs.getInt("idinforme"));
            pel.setData(rs.getString("data"));
            pel.setTitolinforme(rs.getString("titolinforme"));
            pel.setLkinforme(rs.getString("lkinforme"));
            pel.setAlumne(rs.getString("alumne"));
            llista.add(pel);
        }
    } catch (SQLException e) {
        e.printStackTrace();
    }
    cerrarConexion();
    return llista;
}

@Override
public List <String> llistarIdInformes(String nif) {
    List<String> llista = new ArrayList();
    String pel=null;
    abrirConexion();
    try {
        ps=conn.prepareStatement(LLISTAR_IDINFORME);
        ps.setString(1, nif);
        //System.out.println(ps);
        ResultSet rs =ps.executeQuery();
        while(rs.next()){
            pel=rs.getString("idinforme");
            llista.add(pel);
        }
    } catch (SQLException e) {
        e.printStackTrace();
    }
    cerrarConexion();
    return llista;
}

@Override
public Informe cercaInforme(String id) {
    abrirConexion();
    Informe inf = null;
    try {
        ps=conn.prepareStatement(CERCA_INFORME);
        ps.setString(1, id);
        ResultSet rs =ps.executeQuery();
        while(rs.next()){
            inf=new Informe(0, null, null, null, null);
            inf.setIdinforme(rs.getInt("idinforme"));
            inf.setData(rs.getString("data"));
            inf.setTitolinforme(rs.getString("titolinforme"));
            inf.setLkinforme(rs.getString("lkinforme"));
            inf.setAlumne(rs.getString("alumne"));
        }
    } catch (SQLException e) {
        e.printStackTrace();
    }
    cerrarConexion();
    return inf;
}

@Override
public int modificarInforme(String alumne, String lkinforme, String data, String titol, String id) {

```



```

    obrirConexio ();
    int t=0;
    try {
        conn.setAutoCommit(false);
        ps = conn.prepareStatement(MODIFICAR_INFORME);
        ps.setString(1, data);
        ps.setString(2, titol);
        ps.setString(3, lkinforme);
        ps.setString(4, alumne);
        ps.setString(5, id);
        //System.out.println("Operacio: "+ps);
        ps.executeUpdate();
        conn.commit();
        t = 2;
    } catch (SQLException e) {
        try {
            System.out.println("Error, realitzant el rollback: \r\nTipus d'excepció: "+e.toString());
            conn.rollback();
            System.out.println("Rollback realitzat.");
            t = 0;
        } catch (SQLException e1) {
            e1.printStackTrace();
            t = 0;
        }
    }
    tancarConexio();
    return t;
}

// ----- PROFINFORMES -----

@Override
public int crearProfinformes(String numinf, List<String> professional) {
    obrirConexio ();
    int t=0;
    try {
        conn.setAutoCommit(false);
        ps = conn.prepareStatement(CREAR_PROFIFORMES);
        for (int i = 0; i < professional.size(); i++) {
            ps.setString(1, numinf);
            ps.setString(2, professional.get(i));
            ps.addBatch();
        }
        ps.executeBatch();
        conn.commit();
        t = 1;
    } catch (SQLException e) {
        try {
            System.out.println("Error, realitzant el rollback: \r\nTipus d'excepció: "+e.toString());
            conn.rollback();
            System.out.println("Rollback realitzat.");
            t = 0;
        } catch (SQLException e1) {
            e1.printStackTrace();
            t = 0;
        }
    }
    tancarConexio();
    return t;
}

@Override
public int eliminarProfInformes(String id) {
    obrirConexio();
    int t=0;
    try {
        ps=conn.prepareStatement(ELIMINAR_PROFIFORMES);
        ps.setString(1, id);
        ps.executeUpdate();
        t = 3;
    } catch (SQLException e) {
        e.printStackTrace();
        t = 0;
    }
    tancarConexio();
    return t;
}

@Override
public List <Profinformes> llistarProfinformes() {
    List<Profinformes> llista = new ArrayList();
    Profinformes pel=null;
    obrirConexio();
    try {
        ps=conn.prepareStatement(LLISTAR_PROFIFORMES);
        ResultSet rs =ps.executeQuery();
        while(rs.next()){
            pel=new Profinformes(0, null);
            pel.setIdinforme(rs.getInt("idinforme"));
            pel.setNif(rs.getString("nif"));
            llista.add(pel);
        }
    } catch (SQLException e) {
        e.printStackTrace();
    }
    tancarConexio();
    return llista;
}
}

```

```

@Override
public List <String> llistarProfesInformes(String id) {
    obrirConexio();
    List <String> llista = new ArrayList();
    String inf = null;
    try {
        ps=conn.prepareStatement(CERCA_PROFINF);
        ps.setString(1, id);
        ResultSet rs =ps.executeQuery();
        //System.out.println(ps);
        while(rs.next()){
            inf = rs.getString("nif");
            llista.add(inf);
        }
    } catch (SQLException e) {
        e.printStackTrace();
    }
    tancarConexio();
    return llista;
}

@Override
public int modificarProfInformes(String numinf, List<String> professional) {
    obrirConexio ();
    int t=0;
    try {
        conn.setAutoCommit(false);
        ps = conn.prepareStatement(MODIFICAR_PROFIFORMES);
        for (int i = 0; i < professional.size(); i++) {
            ps.setString(1, professional.get(i));
            ps.setString(2, numinf);
            ps.addBatch();
        }
        //System.out.println(ps);
        ps.executeBatch();
        conn.commit();
        t = 2;
    } catch (SQLException e) {
        try {
            System.out.println("Error, realitzant el rollback: \r\nTipus d'exceptiÃ³: "+e.toString());
            conn.rollback();
            System.out.println("Rollback realitzat.");
            t = 0;
        } catch (SQLException e1) {
            e1.printStackTrace();
            t = 0;
        }
    }
    tancarConexio();
    return t;
}

// ----- REUNIONS -----
@Override
public int crearReunio(String datahora, String assistents, String lkcontingut, String propera, String alumne) {
    obrirConexio ();
    int t=0;
    try {
        // Creo la incidència
        conn.setAutoCommit(false);
        ps = conn.prepareStatement(CREAR_REUNIO);
        ps.setString(1, datahora);
        ps.setString(2, assistents);
        ps.setString(3, lkcontingut);
        ps.setString(4, propera);
        ps.setString(5, alumne);
        t = ps.executeUpdate();
        conn.commit();
        // Pillo el nÃºmero de incidència que he creat
        conn.setAutoCommit(false);
        ps = conn.prepareStatement(NUM_REUNIO);
        ResultSet rs =ps.executeQuery();
        while(rs.next()){
            t=Integer.parseInt(rs.getString("maxim"));
        }
        conn.commit();
    } catch (SQLException e) {
        try {
            System.out.println("Error, realitzant el rollback: \r\nTipus d'exceptiÃ³: "+e.toString());
            conn.rollback();
            System.out.println("Rollback realitzat.");
            t = 0;
        } catch (SQLException e1) {
            e1.printStackTrace();
            t = 0;
        }
    }
    tancarConexio();
    return t;
}

@Override
public int eliminarReunio(String id) {
    obrirConexio();
    int t=0;
    try {
        ps=conn.prepareStatement(ELIMINAR_REUNIO);
        ps.setString(1, id);
        ps.executeUpdate();
        t = 3;
    }
}

```

```

    } catch (SQLException e) {
        e.printStackTrace();
        t = 0;
    }
    cerrarConexion();
    return t;
}

@Override
public int modificarReunio(String datahora, String assistents, String lkcontingut,
    String propera, String alumne, String idreunio) {
    abrirConexion ();
    int t=0;
    try {
        conn.setAutoCommit(false);
        ps = conn.prepareStatement(MODIFICAR_REUNIO);
        ps.setString(1, datahora);
        ps.setString(2, assistents);
        ps.setString(3, lkcontingut);
        ps.setString(4, propera);
        ps.setString(5, alumne);
        ps.setString(6, idreunio);
        //System.out.println("Operacio: "+ps);
        ps.executeUpdate();
        conn.commit();
        t = 2;
    } catch (SQLException e) {
        try {
            System.out.println("Error, realitzant el rollback: \r\nTipus d'excepció: "+e.toString());
            conn.rollback();
            System.out.println("Rollback realitzat.");
            t = 0;
        } catch (SQLException e1) {
            e1.printStackTrace();
            t = 0;
        }
    }
    cerrarConexion();
    return t;
}

@Override
public List <String> llistarIdReunions(String nif) {
    List<String> llista = new ArrayList();
    String pel=null;
    abrirConexion();
    try {
        ps=conn.prepareStatement(LLISTAR_IDREUNIO);
        ps.setString(1, nif);
        //System.out.println(ps);
        ResultSet rs =ps.executeQuery();
        //System.out.println(ps);
        while(rs.next()){
            pel=rs.getString("idreunio");
            llista.add(pel);
        }
    } catch (SQLException e) {
        e.printStackTrace();
        //System.out.println(ps);
    }
    cerrarConexion();
    return llista;
}

@Override
public List <Reunio> llistarReunions(String nif) {
    List<Reunio> llista = new ArrayList();
    Reunio pel=null;
    abrirConexion();
    try {
        ps=conn.prepareStatement(LLISTAR_REUNIO);
        ps.setString(1, nif);
        //System.out.println("Cerca: "+ps);
        ResultSet rs =ps.executeQuery();
        while(rs.next()){
            pel=new Reunio(0, null, null, null, 0, null);
            pel.setIdreunio(rs.getInt("idreunio"));
            pel.setDatahora(rs.getString("datahora"));
            pel.setAssistents(rs.getString("assistents"));
            pel.setLkcontingut(rs.getString("lkcontingut"));
            pel.setPropera(rs.getInt("propera"));
            pel.setNif(rs.getString("alumne"));
            llista.add(pel);
        }
    } catch (SQLException e) {
        e.printStackTrace();
    }
    cerrarConexion();
    return llista;
}

@Override
public Reunio cercaReunio(String id) {
    abrirConexion();
    Reunio inf = null;
    try {
        ps=conn.prepareStatement(CERCA_REUNIO);
        ps.setString(1, id);
        ResultSet rs =ps.executeQuery();
    }
}

```

```

        while(rs.next()){
            inf=new Reunio(0, null, null, null, 0, null);
            inf.setIdreunio(rs.getInt("idreunio"));
            inf.setDatahora(rs.getString("datahora"));
            inf.setAssistents(rs.getString("assistents"));
            inf.setLkcontingut(rs.getString("lkcontingut"));
            inf.setPropera(rs.getInt("propera"));
            inf.setNif(rs.getString("alumne"));
        }
    } catch (SQLException e) {
        e.printStackTrace();
    }
    cerrarConexio();
    return inf;
}

// ----- PROFREUNIONS -----
@Override
public int crearProfReunions(String idreunio, List<String> professional) {
    abrirConexio ();
    int t=0;
    try {
        conn.setAutoCommit(false);
        ps = conn.prepareStatement(CREAR_PROFREUNIONS);
        for (int i = 0; i < professional.size(); i++) {
            ps.setString(1, professional.get(i));
            ps.setString(2, idreunio);
            ps.addBatch();
        }
        ps.executeBatch();
        conn.commit();
        t = 1;
    } catch (SQLException e) {
        try {
            System.out.println("Error, realitzant el rollback: \r\nTipus d'excepció: "+e.toString());
            conn.rollback();
            System.out.println("Rollback realitzat.");
            t = 0;
        } catch (SQLException e1) {
            e1.printStackTrace();
            t = 0;
        }
    }
    cerrarConexio();
    return t;
}

@Override
public int eliminarProfReunions(String idreunio) {
    abrirConexio();
    int t=0;
    try {
        ps=conn.prepareStatement(ELIMINAR_PROFREUNIONS);
        ps.setString(1, idreunio);
        ps.executeUpdate();
        t = 3;
    } catch (SQLException e) {
        e.printStackTrace();
        t = 0;
    }
    cerrarConexio();
    return t;
}

@Override
public List <Profreunions> llistarProfReunions() {
    List<Profreunions> llista = new ArrayList();
    Profreunions pel=null;
    abrirConexio();
    try {
        ps=conn.prepareStatement(LLISTAR_PROFREUNIONS);
        ResultSet rs =ps.executeQuery();
        while(rs.next()){
            pel=new Profreunions(null,0);
            pel.setIdreunio(rs.getInt("idreunio"));
            pel.setNif(rs.getString("nif"));
            llista.add(pel);
        }
    } catch (SQLException e) {
        e.printStackTrace();
    }
    cerrarConexio();
    return llista;
}

@Override
public List <String> llistarProfReunions(String id) {
    abrirConexio();
    List <String> llista = new ArrayList();
    String inf = null;
    try {
        ps=conn.prepareStatement(CERCA_PROFREUNIONS);
        ps.setString(1, id);
        ResultSet rs =ps.executeQuery();
        //System.out.println(ps);
        while(rs.next()){
            inf = rs.getString("nif");
            llista.add(inf);
        }
    } catch (SQLException e) {

```

```

        e.printStackTrace();
    }
    cerrarConexion();
    return llista;
}

// ----- FOTOGRAFIES -----
@Override
public int crearFoto(String data, String nomaula, String curs, String lkfoto, String etiquetal,
    String etiqueta2, String etiqueta3, String etiqueta4) {
    abrirConexion ();
    int t=0;
    try {
        conn.setAutoCommit(false);
        ps = conn.prepareStatement(CREAR_FOTO);
        ps.setString(1, lkfoto);
        ps.setString(2, nomaula);
        ps.setString(3, curs);
        ps.setString(4, etiquetal);
        ps.setString(5, etiqueta2);
        ps.setString(6, etiqueta3);
        ps.setString(7, etiqueta4);
        ps.setString(8, data);
        //System.out.println(ps);
        t = ps.executeUpdate();
        conn.commit();
        ps = conn.prepareStatement(NUM_FOTO);
        ResultSet rs =ps.executeQuery();
        while(rs.next()){
            t=Integer.parseInt(rs.getString("maxim"));
        }
    } catch (SQLException e) {
        try {
            System.out.println("Error, realitzant el rollback: \r\nTipus d'excepció: "+e.toString());
            conn.rollback();
            System.out.println("Rollback realitzat.");
            t = 0;
        } catch (SQLException e1) {
            e1.printStackTrace();
            t = 0;
        }
    }
    cerrarConexion();
    return t;
}

@Override
public int eliminarFoto(String id) {
    abrirConexion();
    int t=0;
    try {
        ps=conn.prepareStatement(ELIMINAR_FOTO);
        ps.setString(1, id);
        ps.executeUpdate();
        t = 3;
    } catch (SQLException e) {
        e.printStackTrace();
        t = 0;
    }
    cerrarConexion();
    return t;
}

@Override
public int modificarFoto(String data, String nomaula, String curs, String lkfoto, String
    etiquetal, String etiqueta2, String etiqueta3, String etiqueta4, String id) {
    abrirConexion ();
    int t=0;
    try {
        conn.setAutoCommit(false);
        ps = conn.prepareStatement(MODIFICAR_FOTO);
        ps.setString(1, lkfoto);
        ps.setString(2, nomaula);
        ps.setString(3, curs);
        ps.setString(4, etiquetal);
        ps.setString(5, etiqueta2);
        ps.setString(6, etiqueta3);
        ps.setString(7, etiqueta4);
        ps.setString(8, data);
        ps.setString(9, id);
        //System.out.println("Operacio: "+ps);
        ps.executeUpdate();
        conn.commit();
        t = 2;
    } catch (SQLException e) {
        try {
            System.out.println("Error, realitzant el rollback: \r\nTipus d'excepció: "+e.toString());
            conn.rollback();
            System.out.println("Rollback realitzat.");
            t = 0;
        } catch (SQLException e1) {
            e1.printStackTrace();
            t = 0;
        }
    }
    cerrarConexion();
    return t;
}

@Override

```

```

public List <String> llistarIdFotos() {
    List<String> llista = new ArrayList();
    String pel=null;
    obrirConexio();
    try {
        ps=conn.prepareStatement(LLISTAR_IDFOTOS);
        ResultSet rs =ps.executeQuery();
        //System.out.println(ps);
        while(rs.next()){
            pel=rs.getString("id");
            llista.add(pel);
        }
    } catch (SQLException e) {
        e.printStackTrace();
        System.out.println(ps);
    }
    tancarConexio();
    return llista;
}

@Override
public List <Fotografia> llistarFotos() {
    List<Fotografia> llista = new ArrayList();
    Fotografia pel=null;
    obrirConexio();
    try {
        ps=conn.prepareStatement(LLISTAR_FOTOS);
        ResultSet rs =ps.executeQuery();
        while(rs.next()){
            pel=new Fotografia(null, null, null, null, null, null, null, null, null);
            pel.setId(rs.getInt("id"));
            pel.setData(rs.getString("datafot"));
            pel.setEtiqueta1(rs.getString("etiqueta1"));
            pel.setEtiqueta2(rs.getString("etiqueta2"));
            pel.setEtiqueta3(rs.getString("etiqueta3"));
            pel.setEtiqueta4(rs.getString("etiqueta4"));
            pel.setLkfoto(rs.getString("lkfotografia"));
            pel.setNomaula(rs.getString("nomaula"));
            pel.setCurs(rs.getString("curs"));
            llista.add(pel);
        }
    } catch (SQLException e) {
        e.printStackTrace();
    }
    tancarConexio();
    return llista;
}

@Override
public List <Fotografia> llistarFotosEspecial(String et1, String data1,
    String data2, String aula_nom, String aula_curs, String nif) {
    List<Fotografia> llista = new ArrayList();
    Fotografia pel=null;
    obrirConexio();
    try {
        if (nif==null) {
            if (et1==null) {
                if (data1==null || data2==null) {
                    if (aula_nom==null || aula_curs==null) {
                        ps=conn.prepareStatement(LLISTAR_FOTOS);
                    } else {
                        ps=conn.prepareStatement(LLISTAR_FOTOS_FA);
                        ps.setString(1, aula_nom);
                        ps.setString(2, aula_curs);
                    }
                } else {
                    if (aula_nom==null || aula_curs==null) {
                        ps=conn.prepareStatement(LLISTAR_FOTOS_DA);
                        ps.setString(1, data1);
                        ps.setString(2, data2);
                    } else {
                        ps=conn.prepareStatement(LLISTAR_FOTOS_FA_DA);
                        ps.setString(1, data1);
                        ps.setString(2, data2);
                        ps.setString(3, aula_nom);
                        ps.setString(4, aula_curs);
                    }
                }
            } else {
                if (data1==null || data2==null) {
                    if (aula_nom==null || aula_curs==null) {
                        ps=conn.prepareStatement(LLISTAR_FOTOS_ET1);
                        ps.setString(1, et1);
                        ps.setString(2, et1);
                        ps.setString(3, et1);
                        ps.setString(4, et1);
                    } else {
                        ps=conn.prepareStatement(LLISTAR_FOTOS_FA_ET1);
                        ps.setString(1, aula_nom);
                        ps.setString(2, aula_curs);
                        ps.setString(3, et1);
                        ps.setString(4, et1);
                        ps.setString(5, et1);
                        ps.setString(6, et1);
                    }
                } else {
                    if (aula_nom==null || aula_curs==null) {
                        ps=conn.prepareStatement(LLISTAR_FOTOS_DA_ET1);
                        ps.setString(1, data1);
                    }
                }
            }
        }
    }
}

```



```

        pel.setLkfoto(rs.getString("lkfotografia"));
        pel.setNomaula(rs.getString("nomaula"));
        pel.setCurs(rs.getString("curs"));
        llista.add(pel);
    }
} catch (SQLException e) {
    e.printStackTrace();
}
}
cerrarConexion();
return llista;
}

@Override
public Fotografia cercaFoto(String id) {
    abrirConexion();
    Fotografia inf = null;
    try {
        ps=conn.prepareStatement(CERCA_FOTO);
        ps.setString(1, id);
        ResultSet rs =ps.executeQuery();
        while(rs.next()){
            inf=new Fotografia(null, null, null, null, null, null, null, null, null);
            inf.setId(rs.getInt("id"));
            inf.setData(rs.getString("datafot"));
            inf.setNomaula(rs.getString("nomaula"));
            inf.setCurs(rs.getString("curs"));
            inf.setLkfoto(rs.getString("lkfotografia"));
            inf.setEtiquetal(rs.getString("etiqueta1"));
            inf.setEtiqueta2(rs.getString("etiqueta2"));
            inf.setEtiqueta3(rs.getString("etiqueta3"));
            inf.setEtiqueta4(rs.getString("etiqueta4"));
        }
    } catch (SQLException e) {
        e.printStackTrace();
    }
    cerrarConexion();
    return inf;
}

// ----- FOTOAULUMNES -----
@Override
public int crearFotoalume(String foto, List<String> nif) {
    abrirConexion ();
    int t=0;
    try {
        conn.setAutoCommit(false);
        ps=conn.prepareStatement(ELIMINAR_FOTOALUMNE);
        ps.setString(1, foto);
        ps.executeUpdate();
        ps = conn.prepareStatement(CREAR_FOTOALUMNE);
        for (int i = 0; i < nif.size(); i++) {
            ps.setString(1, nif.get(i));
            ps.setString(2, foto);
            ps.addBatch();
            //System.out.println(ps);
        }
        ps.executeBatch();
        conn.commit();
        t = 1;
    } catch (SQLException e) {
        try {
            System.out.println("Error, realitzant el rollback: \r\nTipus d'excepció: "+e.toString());
            conn.rollback();
            t = 0;
            System.out.println("Rollback realitzat.");
        } catch (SQLException el) {
            el.printStackTrace();
            t = 0;
        }
    }
    cerrarConexion();
    return t;
}

@Override
public int eliminarFotoalume(String id) {
    abrirConexion();
    int t=0;
    try {
        ps=conn.prepareStatement(ELIMINAR_FOTOALUMNE);
        ps.setString(1, id);
        ps.executeUpdate();
        t = 3;
    } catch (SQLException e) {
        e.printStackTrace();
        t = 0;
    }
    cerrarConexion();
    return t;
}

@Override
public List <Fotoalume> llistarFotoalumes() {
    List<Fotoalume> llista = new ArrayList();
    Fotoalume pel=null;
    abrirConexion();
    try {
        ps=conn.prepareStatement(LLISTAR_FOTOALUMNES);
        ResultSet rs =ps.executeQuery();
        while(rs.next()){

```



```

        pel=new Fotoalume(0, null);
        pel.setId(rs.getInt("id"));
        pel.setNif(rs.getString("nif"));
        llista.add(pel);
    }
} catch (SQLException e) {
    e.printStackTrace();
}
}
cerrarConexion();
return llista;
}

@Override
public List <String> cercaFotoalumes(String id) {
    List<String> llista = new ArrayList();
    String pel=null;
    abrirConexion();
    try {
        ps=conn.prepareStatement(LLISTAR_NIFFOTOSALUMNES);
        ps.setString(1, id);
        ResultSet rs =ps.executeQuery();
        //System.out.println(ps);
        while(rs.next()){
            pel=rs.getString("nif");
            llista.add(pel);
        }
    } catch (SQLException e) {
        e.printStackTrace();
        System.out.println(ps);
    }
    cerrarConexion();
    return llista;
}
}

```

CanviFotografia.java

```

/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package servlets;

import java.awt.Graphics2D;
import java.awt.image.BufferedImage;
import java.io.File;
import java.io.IOException;
import java.util.List;
import javax.imageio.ImageIO;
import javax.servlet.ServletException;
import javax.servlet.annotation.MultipartConfig;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;
import jdbc.OperacionsBD;
import org.apache.commons.fileupload.FileItem;
import org.apache.commons.fileupload.disk.DiskFileItemFactory;
import org.apache.commons.fileupload.servlet.ServletFileUpload;

/**
 *
 * @author Carles Torras Font
 */
@WebServlet("/CanviFotografia")
@MultipartConfig
public class CanviFotografia extends HttpServlet {
    private static final long serialVersionUID = 1L;

    // location to store file uploaded
    public static String UPLOAD_DIRECTORY = "arxius" + File.separator + "documents";

    // upload settings
    private static final int MEMORY_THRESHOLD = 1024 * 1024 * 3; // 3MB
    private static final int MAX_FILE_SIZE = 1024 * 1024 * 40; // 40MB
    private static final int MAX_REQUEST_SIZE = 1024 * 1024 * 50; // 50MB
    public static String fileName;

    /**
     * Processes requests for both HTTP <code>GET</code> and <code>POST</code>
     * methods.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    protected void processRequest(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        response.setContentType("text/html;charset=UTF-8");
        request.setCharacterEncoding("UTF-8");
    }

    /**
     * Handles the HTTP <code>POST</code> method.
     */
}

```

```

*
* @param request servlet request
* @param response servlet response
* @throws ServletException if a servlet-specific error occurs
* @throws IOException if an I/O error occurs
*/
@Override
protected void doPost(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {

    // Configuro dades per desar arxius. Uso lib Apache Upload ----->
    DiskFileItemFactory factory = new DiskFileItemFactory();
    // sets memory threshold - beyond which files are stored in disk
    factory.setSizeThreshold(MEMORY_THRESHOLD);
    // sets temporary location to store files
    factory.setRepository(new File(System.getProperty("java.io.tmpdir")));
    ServletFileUpload upload = new ServletFileUpload(factory);
    // sets maximum size of upload file
    upload.setFileSizeMax(MAX_FILE_SIZE);
    // sets maximum size of request (include file + form data)
    upload.setSizeMax(MAX_REQUEST_SIZE);
    // constructs the directory path to store upload file
    // this path is relative to application's directory
    String uploadPath = getServletContext().getRealPath("") + File.separator + UPLOAD_DIRECTORY;
    // creates the directory if it does not exist
    File uploadDir = new File(uploadPath);
    if (!uploadDir.exists()) {
        uploadDir.mkdir();
    }
    // ----->

    // -----> Carrego la info del formulari ----->
    String accio = null;
    try {
        // parses the request's content to extract file data
        @SuppressWarnings("unchecked")
        //List<FileItem> formItems = new ArrayList<FileItem>();
        List<FileItem> formItems = upload.parseRequest(request);
        String arxiu = null;

        if (formItems != null && formItems.size() > 0) {
            // iterates over form's fields

            for (FileItem item : formItems) {
                // processes only fields that are not form fields

                if (!item.isFormField()) {
                    //Agafo l'arxiu
                    fileName = new File(item.getName()).getName();
                    int mida = fileName.length();
                    if (mida > 0) {

                        // Deso la imatge original -> subdirectori "originals"
                        fileName = fileName.replaceAll(" ", "_");
                        File storeFile = new File(getServletContext().getRealPath("") +
                            File.separator + UPLOAD_DIRECTORY + File.separator + "fotosuaris" + File.separator +
                            fileName + ".jpg");

                        item.write(storeFile);

                        // Retoco la imatge i la deso -> subdirectori "modificades"
                        BufferedImage originalImage = ImageIO.read(storeFile);
                        int type = originalImage.getType() == 0? BufferedImage.TYPE_INT_ARGB : originalImage.getType();
                        BufferedImage resizeImageJpg = resizeImage(originalImage, type);
                        storeFile = new File(getServletContext().getRealPath("") +
                            File.separator + UPLOAD_DIRECTORY + File.separator + "fotosuaris" + File.separator +
                            fileName + ".jpg");
                        ImageIO.write(resizeImageJpg, "jpg", storeFile); // deso la imatge en format jpg, amb nom
                        // Ruta per desar al MySQL (foto modificada)
                        arxiu = UPLOAD_DIRECTORY + File.separator + "fotosuaris" + File.separator + fileName + ".jpg";
                        arxiu = arxiu.replaceAll("\\\\", "////////");
                    }
                }
            }
        }

        // Recupero nom d'usuari de la sessió
        HttpSession session = request.getSession();
        String usuari_sessio = (String)session.getAttribute("usuari");
        OperacionsBD op = new OperacionsBD();
        int registre=op.canviarFotoUsuari(usuari_sessio, arxiu);
    } catch (Exception ex) {
        request.setAttribute("message",
            "Hi ha hagut un error: " + ex.getMessage());
    }
}

private static BufferedImage resizeImage(BufferedImage originalImage, int type){
    int nouheight = 91; // mida de la imatge desada
    double width = originalImage.getWidth();
    double height = originalImage.getHeight();
    Double nou = (height / height) * nouheight;
    int nouwidth = nou.intValue();
    BufferedImage resizedImage = new BufferedImage(nouwidth, nouheight, type);
    Graphics2D g = resizedImage.createGraphics();
    g.drawImage(originalImage, 0, 0, nouwidth, nouheight, null);
    g.dispose();
    return resizedImage;
}

```

```
}
}
```

Descarrega.java

```

/**
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package servlets;

import beans.Acces;
import beans.Accesals;
import java.io.File;
import java.io.IOException;
import java.util.List;
import java.util.logging.Level;
import java.util.logging.Logger;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import jdbc.OpRegistre;
import jxl.Workbook;
import jxl.write.Label;
import jxl.write.Number;
import jxl.write.WritableSheet;
import jxl.write.WritableWorkbook;
import jxl.write.WriteException;

/**
 *
 * @author Carles Torras Font
 */
@WebServlet(name = "Descarrega", urlPatterns = {"/Descarrega"})
public class Descarrega extends HttpServlet {

    /**
     * Processes requests for both HTTP <code>GET</code> and <code>POST</code>
     * methods.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    protected void processRequest(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        response.setContentType("text/html;charset=UTF-8");
    }

    /**
     * Handles the HTTP <code>POST</code> method.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    @Override
    protected void doPost(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {

        // Recupero llista d'accessos
        OpRegistre op = new OpRegistre();
        List<Acces> llista = op.llistarTotRegistre();
        OpRegistre op2 = new OpRegistre();
        List<Accesals> llista2 = op2.llistarAccesAl();

        String arxiu = getServletContext().getRealPath("") + File.separator + "arxius" + File.separator + "Accessos.xls";
        String arxiusimpli = "arxius" + File.separator + "Accessos.xls";
        //System.out.println(arxiu);
        WritableWorkbook workbook = Workbook.createWorkbook(new File(arxiu));
        WritableSheet sheet = workbook.createSheet("Accessos", 0);
        Label label1 = new Label(0, 0, "Id AccÃ@s");
        Label label2 = new Label(1, 0, "NIF");
        Label label3 = new Label(2, 0, "Acció");
        Label label4 = new Label(3, 0, "Data/Hora");
        Label label5 = new Label(4, 0, "NIF alumnes");
        try {
            sheet.addCell(label1); sheet.addCell(label2);
            sheet.addCell(label3); sheet.addCell(label4); sheet.addCell(label5);
            sheet.setColumnView(0, 9); sheet.setColumnView(1, 11);
            sheet.setColumnView(2, 25); sheet.setColumnView(3, 25);
            sheet.setColumnView(4, 11);
        } catch (WriteException ex) {
            Logger.getLogger(Descarrega.class.getName()).log(Level.SEVERE, null, ex);
        }
        Number number;
        int comptador;
        for (int i = 0; i < llista.size(); i++) {
            number = new Number(0, i+1, llista.get(i).getIdacces());
            label2 = new Label(1, i+1, llista.get(i).getNif());
            label3 = new Label(2, i+1, llista.get(i).getNomaccio());
        }
    }
}

```

```

label4 = new Label(3, i+1, llista.get(i).getAccesdh());
try {
    sheet.addCell(number); sheet.addCell(label2);
    sheet.addCell(label3); sheet.addCell(label4);
} catch (WriteException ex) {
    Logger.getLogger(Descarrega.class.getName()).log(Level.SEVERE, null, ex);
}

comptador = 0;
for (int j = 0; j < llista2.size(); j++) {
    if (llista2.get(j).getIdacces() == llista.get(i).getIdacces()) {
        comptador = comptador + 1;
        Label etiqueta = new Label(3+comptador, i+1, llista2.get(j).getNif());
        try {
            sheet.setColumnView(3+comptador, 11);
            sheet.addCell(etiqueta);
        } catch (WriteException ex) {
            Logger.getLogger(Descarrega.class.getName()).log(Level.SEVERE, null, ex);
        }
    }
}
workbook.write();
try {
    workbook.close();
} catch (WriteException ex) {
    Logger.getLogger(Descarrega.class.getName()).log(Level.SEVERE, null, ex);
}

response.getWriter().write(arxiusimpli);
processRequest(request, response);
}
}

```

GestorAlumnes.java

```

/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package servlets;

import beans.Alumne;
import beans.Municipi;
import java.io.File;
import java.io.IOException;
import java.util.ArrayList;
import java.util.List;
import javax.servlet.ServletException;
import javax.servlet.annotation.MultipartConfig;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;
import jdbc.OpRegistre;
import jdbc.OperacionsBD;
import org.apache.commons.fileupload.FileItem;
import org.apache.commons.fileupload.disk.DiskFileItemFactory;
import org.apache.commons.fileupload.servlet.ServletFileUpload;

/**
 *
 * @author Carles Torras Font
 */
@WebServlet("/GestorAlumnes")
@MultipartConfig
public class GestorAlumnes extends HttpServlet {
    private static final long serialVersionUID = 1L;

    // location to store file uploaded
    public static String UPLOAD_DIRECTORY = "arxius" + File.separator + "documents";

    // upload settings
    private static final int MEMORY_THRESHOLD = 1024 * 1024 * 3; // 3MB
    private static final int MAX_FILE_SIZE = 1024 * 1024 * 40; // 40MB
    private static final int MAX_REQUEST_SIZE = 1024 * 1024 * 50; // 50MB
    public static String fileName;

    /**
     * Processes requests for both HTTP <code>GET</code> and <code>POST</code>
     * methods.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    protected void processRequest(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        response.setContentType("text/html;charset=UTF-8");
        request.setCharacterEncoding("UTF-8");
    }
}

```

```

* Handles the HTTP <code>POST</code> method.
*
* @param request servlet request
* @param response servlet response
* @throws ServletException if a servlet-specific error occurs
* @throws IOException if an I/O error occurs
*/
@Override
protected void doPost(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {

    // Configuro dades per desar arxius. Uso lib Apache Upload ----->
    DiskFileItemFactory factory = new DiskFileItemFactory();
    // sets memory threshold - beyond which files are stored in disk
    factory.setSizeThreshold(MEMORY_THRESHOLD);
    // sets temporary location to store files
    factory.setRepository(new File(System.getProperty("java.io.tmpdir")));
    ServletFileUpload upload = new ServletFileUpload(factory);
    // sets maximum size of upload file
    upload.setFileSizeMax(MAX_FILE_SIZE);
    // sets maximum size of request (include file + form data)
    upload.setSizeMax(MAX_REQUEST_SIZE);
    // constructs the directory path to store upload file
    // this path is relative to application's directory
    String uploadPath = getServletContext().getRealPath("") + File.separator + UPLOAD_DIRECTORY;
    // creates the directory if it does not exist
    File uploadDir = new File(uploadPath);
    if (!uploadDir.exists()) {
        uploadDir.mkdir();
    }
    // ----->

    // ----- Carrego la info del formulari ----->
    String accio = null;
    try {
        // parses the request's content to extract file data
        @SuppressWarnings("unchecked")
        //List<FileItem> formItems = new ArrayList<FileItem>();
        List<FileItem> formItems = upload.parseRequest(request);
        String arxiu = null;
        String programa = null;
        String nom = null;
        String cognom1 = null;
        String cognom2 = null;
        String nif = null;
        String nifg = null;
        String datanaix = null;
        String entrada = null;
        String adreca = null;
        String poblacio = null;
        String tel1 = null;
        String tel2 = null;
        String tel3 = null;
        String tel4 = null;
        String med1 = null;
        String med2 = null;
        String med3 = null;
        String med4 = null;
        String med5 = null;
        String diag1 = null;
        String diag2 = null;
        String diag3 = null;
        String diag4 = null;
        String diag5 = null;
        String compartit = null;
        String pare = null;
        String mare = null;
        String email = null;
        String numtarsan = null;
        String parellapare = null;
        String parellamare = null;
        String nomtutor = null;
        int idmun = 0;
        //System.out.println("Control0");

        if (formItems != null && formItems.size() > 0) {
            // iterates over form's fields
            //System.out.println("Control01");
            for (FileItem item : formItems) {
                // processes only fields that are not form fields
                //System.out.println(item.getFieldName());
                if (!item.isFormField()) {
                    //Agafo l'arxiu
                    fileName = new File(item.getName()).getName();
                    int mida = fileName.length();
                    if (mida > 0) {
                        fileName = fileName.replaceAll(" ", "_");
                        File storeFile = new File(getServletContext().getRealPath("") +
                            File.separator + UPLOAD_DIRECTORY + File.separator + "programa" + File.separator +
                                fileName);
                        item.write(storeFile); // deso l'arxiu original
                        arxiu = UPLOAD_DIRECTORY + File.separator + "programa" + File.separator + fileName;
                        arxiu = arxiu.replaceAll("\\\\", "/");
                    }
                } else {
                    // Deso les informacions
                    if ("nom".equals(item.getFieldName())) {
                        nom = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
                    } else if ("cognom1".equals(item.getFieldName())) {

```

```

        cognom1 = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
    } else if ("cognom2".equals(item.getFieldName())) {
        cognom2 = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
    } else if ("nifa".equals(item.getFieldName())) {
        nif = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
    } else if ("nifg".equals(item.getFieldName())) {
        nifg = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
        nifg = nifg.substring(Math.max(nifg.length() - 9, 0));
    } else if ("datanaix".equals(item.getFieldName())) {
        datanaix = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
        //System.out.println(datanaix);
    } else if ("entrada".equals(item.getFieldName())) {
        entrada = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
    } else if ("pare".equals(item.getFieldName())) {
        pare = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
    } else if ("mare".equals(item.getFieldName())) {
        mare = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
    } else if ("numtarsan".equals(item.getFieldName())) {
        numtarsan = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
    } else if ("parellapare".equals(item.getFieldName())) {
        parellapare = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
    } else if ("parellamare".equals(item.getFieldName())) {
        parellamare = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
    } else if ("nomtutor".equals(item.getFieldName())) {
        nomtutor = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
    } else if ("diag1".equals(item.getFieldName())) {
        diag1 = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
    } else if ("diag2".equals(item.getFieldName())) {
        diag2 = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
    } else if ("diag3".equals(item.getFieldName())) {
        diag3 = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
    } else if ("diag4".equals(item.getFieldName())) {
        diag4 = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
    } else if ("diag5".equals(item.getFieldName())) {
        diag5 = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
    } else if ("med1".equals(item.getFieldName())) {
        med1 = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
    } else if ("med2".equals(item.getFieldName())) {
        med2 = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
    } else if ("med3".equals(item.getFieldName())) {
        med3 = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
    } else if ("med4".equals(item.getFieldName())) {
        med4 = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
    } else if ("med5".equals(item.getFieldName())) {
        med5 = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
    } else if ("tell".equals(item.getFieldName())) {
        tell = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
    } else if ("tel2".equals(item.getFieldName())) {
        tel2 = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
    } else if ("tel3".equals(item.getFieldName())) {
        tel3 = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
    } else if ("tel4".equals(item.getFieldName())) {
        tel4 = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
    } else if ("adreca".equals(item.getFieldName())) {
        adreca = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
    } else if ("poblacio".equals(item.getFieldName())) {
        poblacio = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
        // Cerco la poblacio
        OperacionsBD op2 = new OperacionsBD();
        idmun=op2.cercarMunicipi(poblacio);
    } else if ("email".equals(item.getFieldName())) {
        email = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
        // Aquest següent només s'usa quan modifiquem arxiu:
    } else if ("programa".equals(item.getFieldName())) {
        programa = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
    } else if ("optionsRadios1".equals(item.getFieldName())) {
        compartit = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
        System.out.println(compartit);
    } else if ("optionsRadios2".equals(item.getFieldName())) {
        accio = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
        //System.out.println("Controll");
    }
}
}
}

// Recupero nom d'usuari de la sessió
HttpSession session = request.getSession();
String usuari_sessio = (String)session.getAttribute("usuari");

// Si afegeixo, tindrà un arxiu no null, si modifico, si tinc el null em quedo amb
// el que tenia, i si no tinc null, em quedo amb el nou (que llavors ja serà arxiu
if (arxiu==null) arxiu = programa;

//System.out.println("Acció: "+accio);
if ("afegir".equals(accio)) {
    OperacionsBD op = new OperacionsBD();
    int registre=op.crearAlumne(new Alumne(nif, compartit, entrada, datanaix, pare, mare,
        adreca, email, tell, tel2, tel3, tel4, diag1, diag5, diag3, diag4, diag5,
        med1, med2, med3, med4, med5, idmun, nom, cognom1, cognom2, arxiu, numtarsan,
        parellapare, parellamare, nomtutor));
    request.setAttribute("informacio", "Alumne inserit correctament. Gràcies.");
    // Deso registre
    List <String> nifs = new ArrayList<>();
    nifs.add(nif);
    OpRegistre or = new OpRegistre(); int i = or.nouRegistre("afegir_alumne",usuari_sessio,nifs);
    request.setAttribute("resultat", registre);
    getServletContext().getRequestDispatcher("/WEB-INF/gestor/preindex.jsp").forward(request, response);
} else if ("modificar".equals(accio)) {

```

```

OperacionsBD op = new OperacionsBD();
Alumne alumne=op.cercaAlumne(nifg);
request.setAttribute("alumne", alumne);
//Recupero llista de municipis
List<String> municipis = new ArrayList<>();
List<Municipi> totmunicipis = new ArrayList<>();
OperacionsBD op5 = new OperacionsBD();
municipis = op5.llistarMunicipis();
request.setAttribute("municipis", municipis);
OperacionsBD op55 = new OperacionsBD();
totmunicipis = op55.llistarTotMunicipis();
// Busco el nom del municipi en relació al codi de l'alumne
String municipi = null;
for (int i = 0; i < municipis.size(); i++) {
    if (totmunicipis.get(i).getIdmun()==alumne.getIdmun()) municipi=totmunicipis.get(i).getNomMunicipi();
}
request.setAttribute("municipi", municipi);

getServletContext().getRequestDispatcher("/WEB-INF/gestor/Modificara.jsp").forward(request, response);
} else if ("modificarex".equals(accio)) {
OperacionsBD op = new OperacionsBD();
//System.out.println("idmun: "+idmun);
int registre=op.modificarAlumne(new Alumne(nif, compartit, entrada, datanaix, pare, mare,
    adreca, email, tel1, tel2, tel3, tel4, diag1, diag2, diag3, diag4, diag5,
    med1, med2, med3, med4, med5, idmun, nom, cognom1, cognom2, arxiu, numtarsan,
    parellapare, parellamare, nomtutor));
request.setAttribute("informacio", "Alumne modificat correctament. Gràcies.");
// Deso registre
List <String> nifs2 = new ArrayList<>();
nifs2.add(nif);
OpRegistre or = new OpRegistre(); int i = or.nouRegistre("modificar_alumne",usuari_sessio,nifs2);
request.setAttribute("resultat", registre);
getServletContext().getRequestDispatcher("/WEB-INF/gestor/preindex.jsp").forward(request, response);
} else if ("eliminar".equals(accio)) {
OperacionsBD op = new OperacionsBD();
int registre=op.eliminarAlumne(nifg);
request.setAttribute("informacio", "Alumne eliminat correctament. Gràcies.");
// Deso registre
List <String> nifs3 = new ArrayList<>();
nifs3.add(nifg);
OpRegistre or = new OpRegistre(); int i = or.nouRegistre("eliminar_alumne",usuari_sessio,nifs3);
request.setAttribute("resultat", registre);
getServletContext().getRequestDispatcher("/WEB-INF/gestor/preindex.jsp").forward(request, response);
} else if ("llistar".equals(accio)) {
OperacionsBD op = new OperacionsBD();
List<Alumne> llistaa = op.cercaAlumne();
request.setAttribute("llistaa", llistaa);

// Deso el registre
List<String> nifs4 = new ArrayList<String>();
for (int i = 0; i < llistaa.size(); i++) {
    String elnifdara = llistaa.get(i).getNif();
    nifs4.add(elnifdara);
}
OpRegistre or = new OpRegistre(); int i = or.nouRegistre("llistar_alumne",usuari_sessio,nifs4);
getServletContext().getRequestDispatcher("/WEB-INF/gestor/Llistaa.jsp").forward(request, response);
}
} catch (Exception ex) {
request.setAttribute("message",
    "Hi ha hagut un error: " + ex.getMessage());
}
}
}
}

```

GestorAssistencia.java

```

/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package servlets;

import beans.Assistencia;
import java.io.File;
import java.io.IOException;
import java.util.ArrayList;
import java.util.List;
import javax.servlet.ServletException;
import javax.servlet.annotation.MultipartConfig;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;
import jdbc.OpRegistre;
import jdbc.OperacionsBD;
import org.apache.commons.fileupload.FileItem;
import org.apache.commons.fileupload.disk.DiskFileItemFactory;
import org.apache.commons.fileupload.servlet.ServletFileUpload;

/**
 *
 * @author Carles Torras Font
 */
@WebServlet("/GestorAssistencia")

```

```

@MultipartConfig
public class GestorAssistencia extends HttpServlet {
    private static final long serialVersionUID = 1L;

    // location to store file uploaded
    public static String UPLOAD_DIRECTORY = "arxius";

    // upload settings
    private static final int MEMORY_THRESHOLD = 1024 * 1024 * 3; // 3MB
    private static final int MAX_FILE_SIZE = 1024 * 1024 * 40; // 40MB
    private static final int MAX_REQUEST_SIZE = 1024 * 1024 * 50; // 50MB
    public static String fileName;

    /**
     * Processes requests for both HTTP <code>GET</code> and <code>POST</code>
     * methods.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    protected void processRequest(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        response.setContentType("text/html;charset=UTF-8");
        request.setCharacterEncoding("UTF-8");
    }

    /**
     * Handles the HTTP <code>POST</code> method.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    @Override
    protected void doPost(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {

        // Configuro dades per desar arxius. Uso lib Apache Upload ----->
        DiskFileItemFactory factory = new DiskFileItemFactory();
        // sets memory threshold - beyond which files are stored in disk
        factory.setSizeThreshold(MEMORY_THRESHOLD);
        // sets temporary location to store files
        factory.setRepository(new File(System.getProperty("java.io.tmpdir")));
        ServletFileUpload upload = new ServletFileUpload(factory);
        // sets maximum size of upload file
        upload.setFileSizeMax(MAX_FILE_SIZE);
        // sets maximum size of request (include file + form data)
        upload.setSizeMax(MAX_REQUEST_SIZE);
        // constructs the directory path to store upload file
        // this path is relative to application's directory
        String uploadPath = getServletContext().getRealPath("") + File.separator + UPLOAD_DIRECTORY;
        // creates the directory if it does not exist
        File uploadDir = new File(uploadPath);
        if (!uploadDir.exists()) {
            uploadDir.mkdir();
        }
        // ----->

        // -----> Carrego la info del formulari ----->
        String accio = null;
        try {
            // parses the request's content to extract file data
            @SuppressWarnings("unchecked")
            List<FileItem> formItems = new ArrayList<FileItem>();
            List<FileItem> formItems = upload.parseRequest(request);
            String alumne = null;
            String motiu = null;
            String data = null;
            String arxiu = null;
            String id = null;
            List<String> alclasse = new ArrayList<>();

            if (formItems != null && formItems.size() > 0) {
                // iterates over form's fields
                //System.out.println("Control01");
                for (FileItem item : formItems) {
                    // processes only fields that are not form fields
                    //System.out.println(item.getFieldName());
                    if (!item.isFormField()) {
                        //Agafo l'arxiu
                        fileName = new File(item.getName()).getName();
                        int mida = fileName.length();
                        if (mida > 0) {
                            fileName = fileName.replaceAll(" ", "_");
                            File storeFile = new File(getServletContext().getRealPath("") +
                                File.separator + UPLOAD_DIRECTORY + File.separator + fileName);
                            item.write(storeFile); // deso l'arxiu original
                            arxiu = UPLOAD_DIRECTORY + File.separator + fileName;
                            arxiu = arxiu.replaceAll("\\\\", "////////");
                        }
                    } else {
                        // Deso les informacions
                        if ("alumnes".equals(item.getFieldName())) {
                            alumne = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
                            alumne = alumne.substring(Math.max(alumne.length() - 9, 0));
                        } else if ("motiu".equals(item.getFieldName())) {

```



```

        motiu = new String(item.getString().getBytes("ISO-8859-1"),"UTF-8");
    } else if ("optionsRadios2".equals(item.getFieldName())) {
        accio = new String(item.getString().getBytes("ISO-8859-1"),"UTF-8");
        //System.out.println("Controll");
    } else if ("data".equals(item.getFieldName())) {
        data = new String(item.getString().getBytes("ISO-8859-1"),"UTF-8");
        //System.out.println("Controll");
    } else if ("id".equals(item.getFieldName())) {
        id = new String(item.getString().getBytes("ISO-8859-1"),"UTF-8");
        //System.out.println("Controll");
    }
    }
}
}
HttpSession session = request.getSession();
String professional = (String)session.getAttribute("usuari");
String rol = (String)session.getAttribute("rol");

//System.out.println("Acció: "+accio);
if ("afegir".equals(accio)) {
    OperacionsBD op = new OperacionsBD();
    int registre=op.crearAssistencia(data, motiu, alumne, professional);
    request.setAttribute("informacio", "AssistÃncia inserida correctament. Gràcies.");
    // Deso registre
    List <String> nifs = new ArrayList<>();
    nifs.add(alumne);
    OpRegistre or = new OpRegistre(); int i = or.nouRegistre("afegir_assist",professional,nifs);
    request.setAttribute("resultat", registre);
    getServletContext().getRequestDispatcher("/WEB-INF/gestor/preindex.jsp").forward(request, response);
} else if ("eliminar".equals(accio)) {
    OperacionsBD op = new OperacionsBD();
    int registre=op.eliminarAssistencia(id);
    //System.out.println(aula_nom+" "+aula_curs);
    request.setAttribute("informacio", "AssistÃncia eliminada correctament. Gràcies.");
    // Deso registre
    OpRegistre or = new OpRegistre(); int i = or.nouRegistre("eliminar_assist",professional,null);
    request.setAttribute("resultat", registre);
    getServletContext().getRequestDispatcher("/WEB-INF/gestor/preindex.jsp").forward(request, response);
} else if ("llistar".equals(accio)) {
    // Llisto les diverses assistencies
    OperacionsBD op = new OperacionsBD();
    List<Assistencia> llistaa = new ArrayList<>();
    if ("Director/a".equals(rol) || "Administrador/a".equals(rol)) {
        llistaa = op.llistarAssistencia("%");
    } else {
        llistaa = op.llistarAssistencia(professional);
    }
    request.setAttribute("llistaa", llistaa);

    // Converteixo NIF en lo llegible (nom, cognom, nif)
    List<String> profes = new ArrayList<>();
    for (int i = 0; i < llistaa.size(); i++) {
        OperacionsBD opl = new OperacionsBD();
        String lotio = opl.cercaProfessional((llistaa.get(i).getProfessional()).toString());
        profes.add(lotio);
    }
    request.setAttribute("nomsprofes", profes);

    List<String> alumnes = new ArrayList<>();
    for (int i = 0; i < llistaa.size(); i++) {
        OperacionsBD opl = new OperacionsBD();
        String lotio = opl.cercaAlumne((llistaa.get(i).getAlumne()).toString());
        alumnes.add(lotio);
    }
    request.setAttribute("nomsalumnes", alumnes);

    // Llista d'alumnes del profe (només podrÃ modificar o eliminar el seus)
    List<String> nifs = new ArrayList<>();
    OperacionsBD op4 = new OperacionsBD();
    nifs = op4.NifAlumnesProf(professional);
    request.setAttribute("alumnesprofe", nifs);

    // Deso registre
    OpRegistre or = new OpRegistre(); int i = or.nouRegistre("llistar_assist",professional,null);
    getServletContext().getRequestDispatcher("/WEB-INF/gestor/Llistaassist.jsp").forward(request, response);
}
} catch (Exception ex) {
    request.setAttribute("message",
        "Hi ha hagut un error: " + ex.getMessage());
}
}
}
}

```

GestorAules.java

```

/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package servlets;

import beans.Classes;
import java.io.File;
import java.io.IOException;
import java.util.ArrayList;

```

```

import java.util.List;
import javax.servlet.ServletException;
import javax.servlet.annotation.MultipartConfig;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;
import jdbc.OpRegistre;
import jdbc.OperacionsBD;
import org.apache.commons.fileupload.FileItem;
import org.apache.commons.fileupload.disk.DiskFileItemFactory;
import org.apache.commons.fileupload.servlet.ServletFileUpload;

/**
 *
 * @author Carles Torras Font
 */
@WebServlet("/GestorAules")
@MultipartConfig
public class GestorAules extends HttpServlet {
    private static final long serialVersionUID = 1L;

    // location to store file uploaded
    public static String UPLOAD_DIRECTORY = "arxius" + File.separator + "documents";

    // upload settings
    private static final int MEMORY_THRESHOLD = 1024 * 1024 * 3; // 3MB
    private static final int MAX_FILE_SIZE = 1024 * 1024 * 40; // 40MB
    private static final int MAX_REQUEST_SIZE = 1024 * 1024 * 50; // 50MB
    public static String fileName;

    /**
     * Processes requests for both HTTP <code>GET</code> and <code>POST</code>
     * methods.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    protected void processRequest(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        response.setContentType("text/html;charset=UTF-8");
        request.setCharacterEncoding("UTF-8");
    }

    /**
     * Handles the HTTP <code>POST</code> method.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    @Override
    protected void doPost(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {

        // Configuro dades per desar arxius. Uso lib Apache Upload ----->
        DiskFileItemFactory factory = new DiskFileItemFactory();
        // sets memory threshold - beyond which files are stored in disk
        factory.setSizeThreshold(MEMORY_THRESHOLD);
        // sets temporary location to store files
        factory.setRepository(new File(System.getProperty("java.io.tmpdir")));
        ServletFileUpload upload = new ServletFileUpload(factory);
        // sets maximum size of upload file
        upload.setFileSizeMax(MAX_FILE_SIZE);
        // sets maximum size of request (include file + form data)
        upload.setSizeMax(MAX_REQUEST_SIZE);
        // constructs the directory path to store upload file
        // this path is relative to application's directory
        String uploadPath = getServletContext().getRealPath("") + File.separator + UPLOAD_DIRECTORY;
        // creates the directory if it does not exist
        File uploadDir = new File(uploadPath);
        if (!uploadDir.exists()) {
            uploadDir.mkdir();
        }
        // ----->

        // -----> Carrego la info del formulari ----->
        String accio = null;
        try {
            // parses the request's content to extract file data
            @SuppressWarnings("unchecked")
            //List<FileItem> formItems = new ArrayList<FileItem>();
            List<FileItem> formItems = upload.parseRequest(request);
            String arxiu = null;
            String nif = null;
            String nomaula = null;
            String curs = null;
            String aula = null;
            String aula_nom = null;
            String aula_curs = null;
            String horariaula = null;
            List<String> alclasse = new ArrayList<>();
            //System.out.println("Control0");

            if (formItems != null && formItems.size() > 0) {

```

```

// iterates over form's fields
//System.out.println("Control01");
for (FormItem item : formItems) {
    // processes only fields that are not form fields
    //System.out.println(item.getFieldName());
    if (!item.isFormField()) {
        //Agafo l'arxiu
        fileName = new File(item.getName().getName());
        int mida = fileName.length();
        if (mida > 0) {
            fileName = fileName.replaceAll(" ", "_");
            File storeFile = new File(getServletContext().getRealPath("") +
                File.separator + UPLOAD_DIRECTORY + File.separator + "horariaula" + File.separator +
                fileName);
            item.write(storeFile); // deso l'arxiu original
            arxiu = UPLOAD_DIRECTORY + File.separator + "horariaula" + File.separator + fileName;
            arxiu = arxiu.replaceAll("\\\\", "/");
        }
    } else {
        // Deso les informacions
        if ("tutor".equals(item.getFieldName())) {
            nif = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
            nif = nif.substring(Math.max(nif.length() - 9, 0));
        } else if ("nom".equals(item.getFieldName())) {
            nomaula = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
        } else if ("optionsRadios2".equals(item.getFieldName())) {
            accio = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
            //System.out.println("Control1");
            // Aquest següent només s'usa quan modifiquem arxiu:
        } else if ("horariaula".equals(item.getFieldName())) {
            horariaula = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
        } else if ("aula".equals(item.getFieldName())) {
            aula = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
            //System.out.println(aula);
            aula_nom = aula.substring(0, Math.max(aula.length() - 11, 0));
            aula_curs = aula.substring(Math.max(aula.length() - 9, 0));
            //System.out.println(aula_nom);
            //System.out.println(aula_curs);
        } else if ("locurs".equals(item.getFieldName())) {
            curs = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
            //System.out.println("Control1");
        }
    }
}
}

// Recupero nom d'usuari de la sessió
HttpSession session = request.getSession();
String usuari_sessio = (String)session.getAttribute("usuari");

// Si afegeixo, tindrà un arxiu no null, si modifico, si tinc el null em quedo amb
// el que tenia, i si no tinc null, em quedo amb el nou (que llavors ja serà arxiu
if (arxiu==null) arxiu = horariaula;

if ("afegir".equals(accio)) {
    OperacionsBD op = new OperacionsBD();
    int registre=op.crearClasse(new Classes(nomaula, curs, arxiu));
    request.setAttribute("informacio", "Aula inserida correctament. Gràcies.");
    // Deso registre
    OpRegistre or = new OpRegistre(); int i = or.nouRegistre("afegir_aula",usuari_sessio,null);
    request.setAttribute("resultat", registre);
    getServletContext().getRequestDispatcher("/WEB-INF/gestor/preindex.jsp").forward(request, response);
} else if ("modificar".equals(accio)) {
    // Per modificar la classe li envio les següents infos:
    // L'aula a modificar
    OperacionsBD op = new OperacionsBD();
    Classes laula=op.cercaClasse(aula_nom, aula_curs);
    request.setAttribute("aula", laula);
    request.setAttribute("nomaula", aula_nom + " " + aula_curs);
    // I crido el JSP de modificació
    getServletContext().getRequestDispatcher("/WEB-INF/gestor/Modificarau.jsp").forward(request, response);
} else if ("modificarex".equals(accio)) {
    OperacionsBD op = new OperacionsBD();
    int registre=op.modificarClasse(new Classes(aula_nom, aula_curs, arxiu));
    //System.out.println(nomaula+" "+curs+" "+arxiu+" "+nif);
    request.setAttribute("informacio", "Aula modificada correctament. Gràcies.");
    // Deso registre
    OpRegistre or = new OpRegistre(); int i = or.nouRegistre("modificar_aula",usuari_sessio,null);
    request.setAttribute("resultat", registre);
    getServletContext().getRequestDispatcher("/WEB-INF/gestor/preindex.jsp").forward(request, response);
} else if ("eliminar".equals(accio)) {
    OperacionsBD op = new OperacionsBD();
    int registre=op.eliminarClasse(aula_nom, aula_curs);
    // Deso registre
    OpRegistre or = new OpRegistre(); int i = or.nouRegistre("eliminar_aula",usuari_sessio,null);
    request.setAttribute("informacio", "Aula eliminada correctament. Gràcies.");
    request.setAttribute("resultat", registre);
    getServletContext().getRequestDispatcher("/WEB-INF/gestor/preindex.jsp").forward(request, response);
} else if ("llistar".equals(accio)) {
    OperacionsBD op = new OperacionsBD();
    List<Classes> llistau = op.llistarClasses();
    request.setAttribute("llistaau", llistau);
    // Deso registre
    OpRegistre or = new OpRegistre(); int i = or.nouRegistre("llistar_aula",usuari_sessio,null);
    getServletContext().getRequestDispatcher("/WEB-INF/gestor/Llistaau.jsp").forward(request, response);
} else if ("assignar".equals(accio)) {
    // ALUMNES -----
    // Llistat d'alumnes total
    OperacionsBD op = new OperacionsBD();
    List<String> llistaalumnes = op.llistarTotsAlumnes();
}

```

```

// Llistat actual de nif's d'alumnes de la classe
OperacionsBD op2 = new OperacionsBD();
List<String> nifsalumnesaula = op2.llistarAlumnesClasse(aula_nom, aula_curs);
// De la llistat de tots els alumnes, pillo els que NIF coincideixi
String partnif;
List<String> alumnesaula = new ArrayList<>();
for (int i = 0; i < llistaalumnes.size(); i++) {
    partnif = llistaalumnes.get(i).substring(Math.max(llibraalumnes.get(i).length() - 9, 0));
    //System.out.println(partnif);
    for (int j = 0; j < nifsalumnesaula.size(); j++) {
        //System.out.println(partnif);
        if (partnif.equals(nifsalumnesaula.get(j))) {
            alumnesaula.add(llibraalumnes.get(i));
        }
    }
}
// I ara els trec de la llista total
for (int i = 0; i < alumnesaula.size(); i++) {
    llistaalumnes.remove(alumnesaula.get(i));
}
request.setAttribute("alumnesaula", alumnesaula);
request.setAttribute("llibraalumnes", llistaalumnes);
request.setAttribute("mida", (2+Math.max(alumnesaula.size(), llistaalumnes.size())));

// PROFES -----
// Llistat de profes total
OperacionsBD op11 = new OperacionsBD();
List<String> llistatuturs = op11.llistarTutors();
// Llistat actual de nif's de tutors de la classe
OperacionsBD op12 = new OperacionsBD();
List<String> nifsprefesaula = op12.llistarProfesClasse(aula_nom, aula_curs);
// De la llistat de tots els tutors, pillo els que NIF coincideixi
String partnif2;
List<String> profesaula = new ArrayList<>();
for (int i = 0; i < llistatuturs.size(); i++) {
    partnif2 = llistatuturs.get(i).substring(Math.max(llibraalumnes.get(i).length() - 9, 0));
    //System.out.println(partnif2);
    for (int j = 0; j < nifsprefesaula.size(); j++) {
        //System.out.println(partnif2);
        if (partnif2.equals(nifsprefesaula.get(j))) {
            profesaula.add(llibraalumnes.get(i));
        }
    }
}
// I ara els trec de la llista total
for (int i = 0; i < profesaula.size(); i++) {
    llistatuturs.remove(profesaula.get(i));
}
request.setAttribute("profesaula", profesaula);
request.setAttribute("llibratuturs", llistatuturs);
request.setAttribute("midap", (2+Math.max(profesaula.size(), llistatuturs.size())));

//Classe que gestiona -----
request.setAttribute("aula_nom", aula_nom);
request.setAttribute("aula_curs", aula_curs);
getServletContext().getRequestDispatcher("/WEB-INF/gestor/Classes.jsp").forward(request, response);
} else if ("nomsaules".equals(accio)) {
    // Recupero els noms de les aules
    OperacionsBD op = new OperacionsBD();
    List<String> llistanomaules = op.llistarNomAules();
    request.setAttribute("llibranomaules", llistanomaules);
    getServletContext().getRequestDispatcher("/WEB-INF/gestor/NomsAules.jsp").forward(request, response);
}
} catch (Exception ex) {
    request.setAttribute("message",
        "Hi ha hagut un error: " + ex.getMessage());
}
}
}
}

```

GestorAules1.java

```

/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package servlets;

import beans.Aula;
import java.io.IOException;
import java.util.ArrayList;
import java.util.List;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;
import jdbc.OpRegistre;
import jdbc.OperacionsBD;

/**
 *
 * @author Carles Torras Font
 */

```

```

*/
@WebServlet("/GestorAules1")
public class GestorAules1 extends HttpServlet {

    /**
     * Processes requests for both HTTP <code>GET</code> and <code>POST</code>
     * methods.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    protected void processRequest(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        response.setContentType("text/html;charset=UTF-8");
        request.setCharacterEncoding("UTF-8");
    }

    /**
     * Handles the HTTP <code>POST</code> method.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    @Override
    protected void doPost(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {

        String accio = new String(request.getParameter("accio").getBytes("ISO-8859-1"), "UTF-8");
        // Recupero nom d'usuari de la sessió
        HttpSession session = request.getSession();
        String usuari_sessio = (String)session.getAttribute("usuari");

        if ("classes".equals(accio)) {
            int registre = 0;
            // ----- Assignar ALUMNES a la classe

            // Carrego la info del formulari ----->
            String nomaula = new String(request.getParameter("nomaula").getBytes("ISO-8859-1"), "UTF-8");
            String curs = new String(request.getParameter("curs").getBytes("ISO-8859-1"), "UTF-8");
            List<String> alclasse = new ArrayList<>();
            String[] resultats = request.getParameterValues("alclasse");

            // Tramito les dades
            if(resultats != null) {
                for(String s : resultats) {
                    alclasse.add(s.substring(Math.max(s.length() - 9, 0)));
                }
                // Considerar que en OperacionsBD quan executo insertarAlumnesClasse
                // primer elimino els que hi han i Després afegeixo aquests
                OperacionsBD op2 = new OperacionsBD();
                registre = op2.insertarAlumnesClasse(nomaula, curs, alclasse);
                //System.out.println("Insertar");
                // Deso registre
                OpRegistre or = new OpRegistre(); int j = or.nouRegistre("assignar_alumnes_aula",usuari_sessio,null);
            } else {
                OperacionsBD op = new OperacionsBD();
                registre = op.eliminarAlumnesClasse(nomaula, curs);
                //System.out.println("Eliminar");
                // Deso registre
                OpRegistre or = new OpRegistre(); int j = or.nouRegistre("assignar_alumnes_aula",usuari_sessio,null);
            }

            // ----- Assignar TUTORS a la classe

            // Carrego la info del formulari ----->
            List<String> tutclasse = new ArrayList<>();
            String[] resultats2 = request.getParameterValues("tutclasse");

            // Tramito les dades
            if(resultats2 != null) {
                for(String s : resultats2) {
                    tutclasse.add(s.substring(Math.max(s.length() - 9, 0)));
                }
                OperacionsBD op2 = new OperacionsBD();
                registre = op2.insertarProfesClasse(nomaula, curs, tutclasse);
                //System.out.println("Insertar");
                // Deso registre
                OpRegistre or = new OpRegistre(); int j = or.nouRegistre("assignar_alumnes_aula",usuari_sessio,null);
            } else {
                OperacionsBD op = new OperacionsBD();
                registre = op.eliminarProfesClasse(nomaula, curs);
                //System.out.println("Eliminar");
                // Deso registre
                OpRegistre or = new OpRegistre(); int j = or.nouRegistre("assignar_alumnes_aula",usuari_sessio,null);
            }
            request.setAttribute("resultat", registre);
            getServletContext().getRequestDispatcher("/WEB-INF/gestor/preindex.jsp").forward(request, response);
        } else if ("nomaules".equals(accio)) {
            // ----- Gestió de nom d'aulas
            // ----- Carrego la info del formulari ----->
            String accio2 = request.getParameter("optionsRadios2");
            int registre = 0;
            if ("afegir".equals(accio2)) {
                String nom = new String(request.getParameter("aulafegir").getBytes("ISO-8859-1"), "UTF-8");
            }
        }
    }
}

```

```

        OperacionsBD op2 = new OperacionsBD();
        registre = op2.crearNomAules(new Aula(nom));
        // Deso registre
        OpRegistre or = new OpRegistre(); int j = or.nouRegistre("afegir_nomaula",usuari_sessio,null);
    } else if ("eliminar".equals(accio2)) {
        String nom = request.getParameter("aulaborrar");
        OperacionsBD op2 = new OperacionsBD();
        registre = op2.eliminarNomAules(nom);
        // Deso registre
        OpRegistre or = new OpRegistre(); int j = or.nouRegistre("eliminar_nomaula",usuari_sessio,null);
    }
    request.setAttribute("resultat", registre);
    getServletContext().getRequestDispatcher("/WEB-INF/gestor/preindex.jsp").forward(request, response);
}
}
}

```

GestorClau.java

```

/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package servlets;

import beans.Aula;
import java.io.IOException;
import java.security.MessageDigest;
import java.security.NoSuchAlgorithmException;
import java.util.ArrayList;
import java.util.List;
import java.util.logging.Level;
import java.util.logging.Logger;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;
import jdbc.OpRegistre;
import jdbc.OperacionsBD;

/**
 *
 * @author Carles Torras Font
 */
@WebServlet("/GestorClau")
public class GestorClau extends HttpServlet {

    /**
     * Processes requests for both HTTP <code>GET</code> and <code>POST</code>
     * methods.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    protected void processRequest(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        response.setContentType("text/html;charset=UTF-8");
        request.setCharacterEncoding("UTF-8");
    }

    /**
     * Handles the HTTP <code>POST</code> method.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    @Override
    protected void doPost(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {

        // Recupero nom d'usuari de la sessió
        HttpSession session = request.getSession();
        String usuari_sessio = (String)session.getAttribute("usuari");

        String clauvella = new String(request.getParameter("clauvella").getBytes("ISO-8859-1"),"UTF-8");
        String claunova = new String(request.getParameter("claunova").getBytes("ISO-8859-1"),"UTF-8");
        String resultat = "0";

        //Comprova clau vella

        // Xifro el password
        StringBuffer hexString = new StringBuffer();
        try {
            MessageDigest md = MessageDigest.getInstance("SHA-256");
            md.update(clauvella.getBytes());
            byte byteData[] = md.digest();
            for (int iiii=0;iiii<byteData.length;iiii++) {
                String hex=Integer.toHexString(0xff & byteData[iiii]);
                if(hex.length()==1) hexString.append('0');
            }
        }
    }
}

```

```

        hexString.append(hex);
    }
} catch (NoSuchAlgorithmException ex) {
    Logger.getLogger(Index.class.getName()).log(Level.SEVERE, null, ex);
}
clauvella = hexString.toString();

// Valido l'usuari
try {
    OperacionsBD op0 = new OperacionsBD();
    List<String> autenticacio = op0.Autenticacio(usuari_sessio);
    // Comprovo si l'autenticaciÃ³ Ã©s bona
    if (autenticacio.get(0).equals(clauvella)) {
        // Xifro el password
        StringBuffer hexString2 = new StringBuffer();
        try {
            MessageDigest md = MessageDigest.getInstance("SHA-256");
            md.update(claunova.getBytes());
            byte byteData[] = md.digest();
            for (int iii=0;iii<byteData.length;iii++) {
                String hex=Integer.toHexString(0xff & byteData[iii]);
                if(hex.length()==1) hexString2.append('0');
                hexString2.append(hex);
            }
        } catch (NoSuchAlgorithmException ex) {
            Logger.getLogger(Index.class.getName()).log(Level.SEVERE, null, ex);
        }
        claunova = hexString2.toString();

        // Canvio el password
        OperacionsBD opl = new OperacionsBD();
        int i = opl.canviPassword(usuari_sessio, claunova);

        resultat = "1";
    }
} catch (Exception e) {
    // Per algun motiu, no he pogut validar l'usuari
    getServletContext().getRequestDispatcher("/index.jsp").forward(request, response);
}

//System.out.println(resultat);
response.getWriter().write(resultat);
processRequest(request, response);
}
}
}

```

GestorFotos.java

```

/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package servlets;

import beans.Fotoalumni;
import beans.Fotografia;
import java.awt.Graphics2D;
import java.awt.image.BufferedImage;
import java.io.File;
import java.io.IOException;
import java.text.DateFormat;
import java.text.SimpleDateFormat;
import java.util.ArrayList;
import java.util.Date;
import java.util.List;
import javax.imageio.ImageIO;
import javax.servlet.ServletException;
import javax.servlet.annotation.MultipartConfig;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;
import jdbc.OpRegistre;
import jdbc.OperacionsBD;
import org.apache.commons.fileupload.FileItem;
import org.apache.commons.fileupload.disk.DiskFileItemFactory;
import org.apache.commons.fileupload.servlet.ServletFileUpload;

/**
 *
 * @author Carles Torras Font
 */
@WebServlet("/GestorFotos")
@MultipartConfig
public class GestorFotos extends HttpServlet {
    private static final long serialVersionUID = 1L;

    // location to store file uploaded
    public static String UPLOAD_DIRECTORY = "arxius" + File.separator + "imatges";

    // upload settings
    private static final int MEMORY_THRESHOLD = 1024 * 1024 * 3; // 3MB
    private static final int MAX_FILE_SIZE = 1024 * 1024 * 40; // 40MB

```

```

private static final int MAX_REQUEST_SIZE = 1024 * 1024 * 50; // 50MB
public static String fileName;

/**
 * Processes requests for both HTTP <code>GET</code> and <code>POST</code>
 * methods.
 *
 * @param request servlet request
 * @param response servlet response
 * @throws ServletException if a servlet-specific error occurs
 * @throws IOException if an I/O error occurs
 */
protected void processRequest(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
    response.setContentType("text/html;charset=UTF-8");
    request.setCharacterEncoding("UTF-8");
}

/**
 * Handles the HTTP <code>POST</code> method.
 *
 * @param request servlet request
 * @param response servlet response
 * @throws ServletException if a servlet-specific error occurs
 * @throws IOException if an I/O error occurs
 */
@Override
protected void doPost(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {

    // Configuro dades per desar arxius. Uso lib Apache Upload ----->
    DiskFileItemFactory factory = new DiskFileItemFactory();
    // sets memory threshold - beyond which files are stored in disk
    factory.setSizeThreshold(MEMORY_THRESHOLD);
    // sets temporary location to store files
    factory.setRepository(new File(System.getProperty("java.io.tmpdir")));
    ServletFileUpload upload = new ServletFileUpload(factory);
    // sets maximum size of upload file
    upload.setFileSizeMax(MAX_FILE_SIZE);
    // sets maximum size of request (include file + form data)
    upload.setSizeMax(MAX_REQUEST_SIZE);
    // constructs the directory path to store upload file
    // this path is relative to application's directory
    String uploadPath = getServletContext().getRealPath("") + File.separator + UPLOAD_DIRECTORY;
    // creates the directory if it does not exist
    File uploadDir = new File(uploadPath);
    if (!uploadDir.exists()) {
        uploadDir.mkdir();
    }
    // ----->

    // -----> Carrego la info del formulari ----->
    String accio = null;
    try {
        // parses the request's content to extract file data
        @SuppressWarnings("unchecked")
        //List<FileItem> formItems = new ArrayList<FileItem>();
        List<FileItem> formItems = upload.parseRequest(request);
        String etiqueta1 = null;
        String etiqueta2 = null;
        String etiqueta3 = null;
        String etiqueta4 = null;
        String data = null;
        String aula = null;
        String aula_nom = null;
        String aula_curs = null;
        String lkfoto = null;
        String arxiu = null;
        String id = null;
        List<String> alumnes = new ArrayList<>();

        if (formItems != null && formItems.size() > 0) {
            // iterates over form's fields

            for (FileItem item : formItems) {
                // processes only fields that are not form fields

                if (!item.isFormField()) {
                    //Agafo l'arxiu
                    fileName = new File(item.getName()).getName();
                    int mida = fileName.length();
                    if (mida > 0) {
                        // Creo el timestamp
                        DateFormat format = new SimpleDateFormat("yyyyMMdd_HHmss");
                        Date diahora = new Date();
                        String ara = format.format(diahora);

                        // Deso la imatge original -> subdirectori "originals"
                        fileName = fileName.replaceAll(" ", "_");
                        File storeFile = new File(getServletContext().getRealPath("") +
                            File.separator + UPLOAD_DIRECTORY + File.separator + "originals" + File.separator +
                            fileName + "_" + ara + ".jpg");
                        item.write(storeFile);

                        // Retoco la imatge i la deso -> subdirectori "modificades"
                        BufferedImage originalImage = ImageIO.read(storeFile);
                        int type = originalImage.getType() == 0? BufferedImage.TYPE_INT_ARGB : originalImage.getType();
                        BufferedImage resizeImageJpg = resizeImage(originalImage, type);
                        storeFile = new File(getServletContext().getRealPath("") +

```



```

        File.separator + UPLOAD_DIRECTORY + File.separator + "modificades" + File.separator +
fileName + "_mod_" + ara + ".jpg");
        ImageIO.write(resizeImageJpg, "jpg", storeFile); // deso la imatge en format jpg, amb nom
À*nic i mida adequada

        // Ruta per desar al MySQL (foto modificada)
        arxiu = UPLOAD_DIRECTORY + File.separator + "modificades" + File.separator + fileName + "_mod_" +
ara + ".jpg";
        arxiu = arxiu.replaceAll("\\\\", "/////");
    } else {
        // Deso les informacions
        if ("et1".equals(item.getFieldName())) {
            etiquetal = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
        } else if ("et2".equals(item.getFieldName())) {
            etiqueta2 = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
        } else if ("et3".equals(item.getFieldName())) {
            etiqueta3 = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
        } else if ("et4".equals(item.getFieldName())) {
            etiqueta4 = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
        } else if ("id".equals(item.getFieldName())) {
            id = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
        } else if ("data".equals(item.getFieldName())) {
            data = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
            //System.out.println(data);
        } else if ("fotoactual".equals(item.getFieldName())) {
            lkfoto = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
        } else if ("aula".equals(item.getFieldName())) {
            aula = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
            aula_nom = aula.substring(0, Math.max(aula.length() - 11, 0));
            aula_curs = aula.substring(Math.max(aula.length() - 9, 0));
        } else if ("optionsRadios2".equals(item.getFieldName())) {
            accio = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
        } else if ("alumnes".equals(item.getFieldName())) {
            String lalumne = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
            lalumne = lalumne.substring(Math.max(lalumne.length() - 9, 0));
            alumnes.add(lalumne);
        }
    }
}

// Recupero nom d'usuari de la sessió
HttpSession session = request.getSession();
String usuari_sessio = (String)session.getAttribute("usuari");

//System.out.println("Acció: "+accio);
if (arxiu==null) arxiu = lkfoto;
if ("afegir".equals(accio)) {
    // Afegeixo la fotografia
    OperacionsBD op = new OperacionsBD();
    registre=op.crearFoto(data, aula_nom, aula_curs, arxiu, etiquetal, etiqueta2, etiqueta3, etiqueta4);

    // Afegeixo els alumnes associats
    if (alumnes.size() > 0) {
        OperacionsBD op2 = new OperacionsBD();
        registre2=op2.crearFotoalumne(Integer.toString(registre), alumnes);
    }

    request.setAttribute("informacio", "Foto inserida correctament. Gràcies.");
    // Deso registre
    OpRegistre or = new OpRegistre(); int i = or.nouRegistre("afegir_foto",usuari_sessio,null);
    if (registre > 0) registre = 1;
    request.setAttribute("resultat", registre);
    getServletContext().getRequestDispatcher("/WEB-INF/gestor/preindex.jsp").forward(request, response);
} else if ("modificar".equals(accio)) {
    // Foto a modificar
    OperacionsBD op = new OperacionsBD();
    Fotografia foto=op.cercaFoto(id);
    request.setAttribute("fotografia", foto);

    // Alumnes de la foto a modificar
    OperacionsBD op3 = new OperacionsBD();
    List<String> llistafal = op3.cercaFotoalumnes(id);
    request.setAttribute("llistafal", llistafal);

    //Recupero llista d'alumnes
    List<String> elsalumnes = new ArrayList<>();
    OperacionsBD op5 = new OperacionsBD();
    elsalumnes = op5.llistarTotsAlumnes();
    request.setAttribute("elsalumnes", elsalumnes);

    //Agafo els NIFs de la llista d'alumnes
    List<String> elsalumnesnif = new ArrayList<>();
    String alumenenif;
    for (int i=0; i<elsalumnes.size(); i++) {
        alumenenif = elsalumnes.get(i).substring(Math.max(elsalumnes.get(i).length() - 9, 0));
        elsalumnesnif.add(alumenenif);
        //System.out.println(alumenenif);
    }
    request.setAttribute("elsalumnesnif", elsalumnesnif);

    //Recupero llista d'aules
    List<String> aules = new ArrayList<>();
    OperacionsBD opl = new OperacionsBD();
    aules = opl.llistarClasseCurs();
    request.setAttribute("aules", aules);

    String lonomdaula = foto.getNomaula() + ", " + foto.getCurs();

```

```

        request.setAttribute("lonomdaula", lonomdaula);

        getServletContext().getRequestDispatcher("/WEB-INF/gestor/ModificarFotos.jsp").forward(request, response);
    } else if ("modificarex".equals(accio)) {
        // Modifico la foto
        OperacionsBD op = new OperacionsBD();
        int registre=op.modificarFoto(data, aula_nom, aula_curs, arxiu, etiqueta1, etiqueta2, etiqueta3, etiqueta4,
id);

        request.setAttribute("informacio", "Foto modificada correctament. Gràcies.");

        // Modifico els alumnes de la foto
        if (alumnes.size()>0) {
            OperacionsBD op2 = new OperacionsBD();
            int registre2=op2.crearFotoalumne(id, alumnes);
        } else {
            OperacionsBD op1 = new OperacionsBD();
            int registre1=op1.eliminarFotoalumne(id);
        }

        // Deso registre
        OpRegistre or = new OpRegistre(); int i = or.nouRegistre("modificar_foto",usuari_sessio,null);
        request.setAttribute("resultat", registre);
        getServletContext().getRequestDispatcher("/WEB-INF/gestor/preindex.jsp").forward(request, response);
    } else if ("eliminar".equals(accio)) {
        // Primer elimino els alumnes associats...
        OperacionsBD opl = new OperacionsBD();
        int registre1=opl.eliminarFotoalumne(id);
        // I Després la foto
        OperacionsBD op = new OperacionsBD();
        int registre=op.eliminarFoto(id);
        request.setAttribute("informacio", "Foto eliminada correctament. Gràcies.");
        // Deso registre
        OpRegistre or = new OpRegistre(); int i = or.nouRegistre("eliminar_foto",usuari_sessio,null);
        request.setAttribute("resultat", registre);
        getServletContext().getRequestDispatcher("/WEB-INF/gestor/preindex.jsp").forward(request, response);
    } else if ("llistar".equals(accio)) {
        // Recupero catà leg de fotos
        OperacionsBD op = new OperacionsBD();
        List<Fotografia> llistaf = op.llistarFotos();
        request.setAttribute("llistaf", llistaf);
        //Recupero llista de fotos
        List<String> fotos = new ArrayList<>();
        OperacionsBD op27 = new OperacionsBD();
        fotos = op27.llistarIdFotos();
        request.setAttribute("fotos", fotos);
        // Recupero llista d'alumnes per foto, en el JSP ja classificaré
        OperacionsBD op2 = new OperacionsBD();
        List<Fotoalumne> llistafa = op2.llistarFotoalumnes();
        request.setAttribute("llistafa", llistafa);
        //Recupero llista d'aules (per filtrar)
        List<String> aules = new ArrayList<>();
        OperacionsBD opl = new OperacionsBD();
        aules = opl.llistarClasseCurs();
        request.setAttribute("aules", aules);
        //Recupero llista d'alumnes (per filtrar)
        List<String> alumnesfiltre = new ArrayList<>();
        OperacionsBD op5 = new OperacionsBD();
        alumnesfiltre = op5.llistarTotsAlumnes();
        request.setAttribute("alumnesfiltre", alumnesfiltre);
        // Deso registre
        OpRegistre or = new OpRegistre(); int i = or.nouRegistre("llistar_foto",usuari_sessio,null);
        getServletContext().getRequestDispatcher("/WEB-INF/gestor/Llistaf.jsp").forward(request, response);
    }
    } catch (Exception ex) {
        request.setAttribute("message",
            "Hi ha hagut un error: " + ex.getMessage());
    }
}

private static BufferedImage resizeImage(BufferedImage originalImage, int type){
    int nouwidth = 1020; //mida de la imatge desada
    double width = originalImage.getWidth();
    double height = originalImage.getHeight();
    Double nou = (height / width) * nouwidth;
    int nouheight = nou.intValue();
    BufferedImage resizedImage = new BufferedImage(nouwidth, nouheight, type);
    Graphics2D g = resizedImage.createGraphics();
    g.drawImage(originalImage, 0, 0, nouwidth, nouheight, null);
    g.dispose();
    return resizedImage;
}
}

```

GestorFotos2.java

```

/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package servlets;

import beans.Fotoalumne;
import beans.Fotografia;
import java.io.IOException;
import java.util.ArrayList;

```

```

import java.util.List;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;
import jdbc.OpRegistre;
import jdbc.OperacionsBD;

/**
 *
 * @author Carles Torras Font
 */
@WebServlet("/GestorFotos1")
public class GestorFotos1 extends HttpServlet {

    /**
     * Processes requests for both HTTP <code>GET</code> and <code>POST</code>
     * methods.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    protected void processRequest(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        response.setContentType("text/html;charset=UTF-8");
        request.setCharacterEncoding("UTF-8");
    }

    /**
     * Handles the HTTP <code>POST</code> method.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    @Override
    protected void doPost(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {

        String et1 = null;
        String aula_nom = null;
        String aula_curs = null;
        String aula = null;
        String data1 = null;
        String data2 = null;
        String alumne = null;
        String[] filtros = request.getParameterValues("filtres");
        try {
            for (String s : filtros) {
                //System.out.println(s);
                if ("et1c".equals(s)) {
                    et1 = new String(request.getParameter("et1").getBytes("ISO-8859-1"), "UTF-8");
                    if (et1.length() < 1) et1 = null;
                }
                if ("aula".equals(s)) {
                    aula = new String(request.getParameter("laopcio").getBytes("ISO-8859-1"), "UTF-8");
                    aula_nom = aula.substring(0, Math.max(aula.length() - 11, 0));
                    aula_curs = aula.substring(Math.max(aula.length() - 9, 0));
                }
                if ("calendari".equals(s)) {
                    data1 = request.getParameter("ladata1");
                    data2 = request.getParameter("ladata2");
                    if (data1.length() < 1 || data2.length() < 1) {
                        data1 = null;
                        data2 = null;
                    }
                }
                if ("alumne".equals(s)) {
                    alumne = new String(request.getParameter("filtraalumnes").getBytes("ISO-8859-1"), "UTF-8");
                    alumne = alumne.substring(Math.max(alumne.length() - 9, 0));
                    //System.out.println(alumne);
                }
            }
        } catch (NullPointerException e) {}

        // Recupero nom d'usuari de la sessió
        HttpSession session = request.getSession();
        String usuari_sessio = (String) session.getAttribute("usuari");

        OperacionsBD op = new OperacionsBD();
        //System.out.println(et1+data1+data2+aula_nom+aula_curs);
        List<Fotografia> llistaf = op.llistarFotosEspecial(et1, data1, data2, aula_nom, aula_curs, alumne);
        request.setAttribute("l1listaf", llistaf);
        // Recupero llista d'alumnes per foto, en el JSP ja classificaré
        OperacionsBD op2 = new OperacionsBD();
        List<Fotoalumne> llistafa = op2.llistarFotoalumnes();
        request.setAttribute("l1listafa", llistafa);
        //Recupero llista d'ales (per filtrar)
        List<String> aules = new ArrayList<>();
        OperacionsBD opl = new OperacionsBD();
        aules = opl.llistarClasseCurs();
        request.setAttribute("aules", aules);
        //Recupero llista d'alumnes (per filtrar)
        List<String> alumnesfiltre = new ArrayList<>();

```

```

OperacionsBD op5 = new OperacionsBD();
alumnesfiltre = op5.llistarTotsAlumnes();
request.setAttribute("alumnesfiltre", alumnesfiltre);
// Deso registre
OpRegistre or = new OpRegistre(); int i = or.nouRegistre("llistar_foto",usuari_sessio,null);
getServletContext().getRequestDispatcher("/WEB-INF/gestor/Llistaf.jsp").forward(request, response);
}
}

```

GestorIncidencia.java

```

/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package servlets;

import beans.Incialumne;
import beans.Incidencia;
import java.io.File;
import java.io.IOException;
import java.util.ArrayList;
import java.util.List;
import javax.servlet.ServletException;
import javax.servlet.annotation.MultipartConfig;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;
import jdbc.OpRegistre;
import jdbc.OperacionsBD;
import org.apache.commons.fileupload.FileItem;
import org.apache.commons.fileupload.disk.DiskFileItemFactory;
import org.apache.commons.fileupload.servlet.ServletFileUpload;

/**
 *
 * @author Carles Torras Font
 */
@WebServlet("/GestorIncidencia")
@MultipartConfig
public class GestorIncidencia extends HttpServlet {
    private static final long serialVersionUID = 1L;

    // location to store file uploaded
    public static String UPLOAD_DIRECTORY = "arxius" + File.separator + "documents";

    // upload settings
    private static final int MEMORY_THRESHOLD = 1024 * 1024 * 3; // 3MB
    private static final int MAX_FILE_SIZE = 1024 * 1024 * 40; // 40MB
    private static final int MAX_REQUEST_SIZE = 1024 * 1024 * 50; // 50MB
    public static String fileName;

    /**
     * Processes requests for both HTTP GET and POST
     * methods.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    protected void processRequest(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        response.setContentType("text/html;charset=UTF-8");
        request.setCharacterEncoding("UTF-8");
    }

    /**
     * Handles the HTTP POST method.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    @Override
    protected void doPost(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {

        // Configuro dades per desar arxius. Uso lib Apache Upload ----->
        DiskFileItemFactory factory = new DiskFileItemFactory();
        // sets memory threshold - beyond which files are stored in disk
        factory.setSizeThreshold(MEMORY_THRESHOLD);
        // sets temporary location to store files
        factory.setRepository(new File(System.getProperty("java.io.tmpdir")));
        ServletFileUpload upload = new ServletFileUpload(factory);
        // sets maximum size of upload file
        upload.setFileSizeMax(MAX_FILE_SIZE);
        // sets maximum size of request (include file + form data)
        upload.setSizeMax(MAX_REQUEST_SIZE);
        // constructs the directory path to store upload file
        // this path is relative to application's directory

```

```

String uploadPath = getServletContext().getRealPath("") + File.separator + UPLOAD_DIRECTORY;
// creates the directory if it does not exist
File uploadDir = new File(uploadPath);
if (!uploadDir.exists()) {
    uploadDir.mkdir();
}
// ----->

// ----- Carrego la info del formulari ----->
String accio = null;
try {
    // parses the request's content to extract file data
    @SuppressWarnings("unchecked")
    //List<FileItem> formItems = new ArrayList<FileItem>();
    List<FileItem> formItems = upload.parseRequest(request);
    String titol = null;
    String lkinforme = null;
    String data = null;
    String arxiu = null;
    String id = null;
    List<String> alumnes = new ArrayList<>();

    if (formItems != null && formItems.size() > 0) {
        // iterates over form's fields
        //System.out.println("Control01");
        for (FileItem item : formItems) {
            // processes only fields that are not form fields
            //System.out.println(item.getFieldName());
            if (!item.isFormField()) {
                //Agafo l'arxiu
                fileName = new File(item.getName()).getName();
                int mida = fileName.length();
                if (mida > 0) {
                    fileName = fileName.replaceAll(" ", "_");
                    File storeFile = new File(getServletContext().getRealPath("") +
                        File.separator + UPLOAD_DIRECTORY + File.separator + "incidencies" + File.separator +
                            fileName);

                    item.write(storeFile); // deso l'arxiu original
                    arxiu = UPLOAD_DIRECTORY + File.separator + "incidencies" + File.separator + fileName;
                    arxiu = arxiu.replaceAll("\\\\", "/");
                }
            } else {
                // Deso les informacions
                // Aquest següent només s'usa quan modifiquem arxiu:
                if ("lkinforme".equals(item.getFieldName())) {
                    lkinforme = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
                } else if ("titol".equals(item.getFieldName())) {
                    titol = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
                } else if ("optionsRadios2".equals(item.getFieldName())) {
                    accio = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
                    //System.out.println("Control1");
                } else if ("data".equals(item.getFieldName())) {
                    data = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
                    //System.out.println("Control1");
                } else if ("id".equals(item.getFieldName())) {
                    id = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
                    //System.out.println("Control1");
                } else if ("alumnes".equals(item.getFieldName())) {
                    String lalumne = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
                    lalumne = lalumne.substring(Math.max(lalumne.length() - 9, 0));
                    alumnes.add(lalumne);
                }
            }
        }
    }

    HttpSession session = request.getSession();
    String professional = (String)session.getAttribute("usuari");
    String rol = (String)session.getAttribute("rol");

    // Si afegeixo, tindrè un arxiu no null, si modifico, si tinc el null em quedo amb
    // el que tenia, i si no tinc null, em quedo amb el nou (que llavors ja serà arxiu)
    if (arxiu==null) arxiu = lkinforme;

    //System.out.println("Acció: "+accio);
    if ("afegir".equals(accio)) {
        // Creo la incidència
        OperacionsBD op = new OperacionsBD();
        int registre=op.crearIncidencia(professional,arxiu,data,titol);
        // I omplio la taula que afegeix alumnes a la incidència
        OperacionsBD op2 = new OperacionsBD();
        int registre2=op2.crearIncialumne(String.valueOf(registre), alumnes);
        request.setAttribute("informacio", "Incidència inserida correctament. Gràcies.");
        // Deso registre
        OpRegistre or = new OpRegistre(); int i = or.nouRegistre("afegir_incidencia",professional,alumnes);
        request.setAttribute("resultat", registre2);
        getServletContext().getRequestDispatcher("/WEB-INF/gestor/preindex.jsp").forward(request, response);
    } else if ("eliminar".equals(accio)) {
        // Elimino els alumnes de la incidència
        OperacionsBD op2 = new OperacionsBD();
        int registre2=op2.eliminarIncialumne(id);
        // Elimino la incidència
        OperacionsBD op = new OperacionsBD();
        int registre=op.eliminarIncidencia(id);
        request.setAttribute("informacio", "Incidència eliminada correctament. Gràcies.");
        // Deso registre
        OpRegistre or = new OpRegistre(); int i = or.nouRegistre("eliminar_incidencia",professional,null);
        request.setAttribute("resultat", registre);
        getServletContext().getRequestDispatcher("/WEB-INF/gestor/preindex.jsp").forward(request, response);
    } else if ("llistar".equals(accio)) {

```

```

// Agafo dades de la incidència
OperacionsBD op = new OperacionsBD();
List<Incidencia> llistaa = new ArrayList<>();
if ("Director/a".equals(rol) || "Administrador/a".equals(rol)) {
    llistaa = op.llistarIncidencia("%");
} else {
    llistaa = op.llistarIncidencia(professional);
}
request.setAttribute("llistaa", llistaa);

// Converteixo NIF en lo llegible (nom, cognom, nif)
List<String> profes = new ArrayList<>();
for (int i = 0; i < llistaa.size(); i++) {
    OperacionsBD op1 = new OperacionsBD();
    String lotio = op1.cercaProfessional((llistaa.get(i).getNif()).toString());
    profes.add(lotio);
}
request.setAttribute("nomsprofes", profes);

// I dels alumnes involucrats. Després discrimino cada un de quina incidència és
OperacionsBD op2 = new OperacionsBD();
List<Incialumne> llistaal = new ArrayList<>();
if ("Director/a".equals(rol) || "Administrador/a".equals(rol)) {
    llistaal = op2.llistarIncialumne("%");
} else {
    llistaal = op2.llistarIncialumne(professional);
}
request.setAttribute("llistaal", llistaal);

// Converteixo NIF en lo llegible (nom, cognom, nif)
List<String> losalumnes = new ArrayList<>();
for (int i = 0; i < llistaal.size(); i++) {
    OperacionsBD op3 = new OperacionsBD();
    String lotio = op3.cercaAlumne((llistaal.get(i).getNif()).toString());
    losalumnes.add(lotio);
}
request.setAttribute("nomsalumnes", losalumnes);

// LLista d'alumnes del profe (només podrà modificar o eliminar el seus)
List<String> nifs = new ArrayList<>();
OperacionsBD op4 = new OperacionsBD();
nifs = op4.NifAlumnesProf(professional);
request.setAttribute("alumnesprofe", nifs);

// Deso registre
OpRegistre or = new OpRegistre(); int i = or.nouRegistre("llistar_incidencia",professional,null);
getServletContext().getRequestDispatcher("/WEB-INF/gestor/Llistainci.jsp").forward(request, response);
}
} catch (Exception ex) {
    request.setAttribute("message",
        "Hi ha hagut un error: " + ex.getMessage());
}
}
}
}

```

GestorInformes.java

```

/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package servlets;

import beans.Informe;
import beans.Profinformes;
import java.io.File;
import java.io.IOException;
import java.util.ArrayList;
import java.util.List;
import javax.servlet.ServletException;
import javax.servlet.annotation.MultipartConfig;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;
import jdbc.OpRegistre;
import jdbc.OperacionsBD;
import org.apache.commons.fileupload.FileItem;
import org.apache.commons.fileupload.disk.DiskFileItemFactory;
import org.apache.commons.fileupload.servlet.ServletFileUpload;

/**
 *
 * @author Carles Torras Font
 */
@WebServlet("/GestorInformes")
@MultipartConfig
public class GestorInformes extends HttpServlet {
    private static final long serialVersionUID = 1L;

    // location to store file uploaded
    public static String UPLOAD_DIRECTORY = "arxius" + File.separator + "documents";

    // upload settings
    private static final int MEMORY_THRESHOLD = 1024 * 1024 * 3; // 3MB
    private static final int MAX_FILE_SIZE = 1024 * 1024 * 40; // 40MB

```

```

private static final int MAX_REQUEST_SIZE = 1024 * 1024 * 50; // 50MB
public static String fileName;

/**
 * Processes requests for both HTTP <code>GET</code> and <code>POST</code>
 * methods.
 *
 * @param request servlet request
 * @param response servlet response
 * @throws ServletException if a servlet-specific error occurs
 * @throws IOException if an I/O error occurs
 */
protected void processRequest(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
    response.setContentType("text/html;charset=UTF-8");
    request.setCharacterEncoding("UTF-8");
}

/**
 * Handles the HTTP <code>POST</code> method.
 *
 * @param request servlet request
 * @param response servlet response
 * @throws ServletException if a servlet-specific error occurs
 * @throws IOException if an I/O error occurs
 */
@Override
protected void doPost(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {

    // Configuro dades per desar arxius. Uso lib Apache Upload ----->
    DiskFileItemFactory factory = new DiskFileItemFactory();
    // sets memory threshold - beyond which files are stored in disk
    factory.setSizeThreshold(MEMORY_THRESHOLD);
    // sets temporary location to store files
    factory.setRepository(new File(System.getProperty("java.io.tmpdir")));
    ServletFileUpload upload = new ServletFileUpload(factory);
    // sets maximum size of upload file
    upload.setFileSizeMax(MAX_FILE_SIZE);
    // sets maximum size of request (include file + form data)
    upload.setSizeMax(MAX_REQUEST_SIZE);
    // constructs the directory path to store upload file
    // this path is relative to application's directory
    String uploadPath = getServletContext().getRealPath("") + File.separator + UPLOAD_DIRECTORY;
    // creates the directory if it does not exist
    File uploadDir = new File(uploadPath);
    if (!uploadDir.exists()) {
        uploadDir.mkdir();
    }
    // ----->

    // ----- Carrego la info del formulari ----->
    String accio = null;
    try {
        // parses the request's content to extract file data
        @SuppressWarnings("unchecked")
        //List<FileItem> formItems = new ArrayList<FileItem>();
        List<FileItem> formItems = upload.parseRequest(request);
        String titol = null;
        String lkinforme = null;
        String data = null;
        String arxiu = null;
        String id = null;
        String alumne = null;
        List<String> professionals = new ArrayList<>();

        if (formItems != null && formItems.size() > 0) {
            // iterates over form's fields
            //System.out.println("Control01");
            for (FileItem item : formItems) {
                // processes only fields that are not form fields
                //System.out.println(item.getFieldName());
                if (!item.isFormField()) {
                    //Agafo l'arxiu
                    fileName = new File(item.getName()).getName();
                    int mida = fileName.length();
                    if (mida > 0) {
                        fileName = fileName.replaceAll(" ", "_");
                        File storeFile = new File(getServletContext().getRealPath("") +
                            File.separator + UPLOAD_DIRECTORY + File.separator + "informes" + File.separator +
                                fileName);

                        item.write(storeFile); // deso l'arxiu original
                        arxiu = UPLOAD_DIRECTORY + File.separator + "informes" + File.separator + fileName;
                        arxiu = arxiu.replaceAll("\\\\", "/");
                    }
                } else {
                    // Deso les informacions
                    // Aquest següent només s'usa quan modifiquem arxiu:
                    if ("lkinforme".equals(item.getFieldName())) {
                        lkinforme = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
                    } else if ("titol".equals(item.getFieldName())) {
                        titol = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
                    } else if ("optionsRadios2".equals(item.getFieldName())) {
                        accio = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
                    } else if ("data".equals(item.getFieldName())) {
                        data = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
                    } else if ("id".equals(item.getFieldName())) {
                        id = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
                    } else if ("alumnes".equals(item.getFieldName())) {

```



```

import javax.servlet.annotation.MultipartConfig;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;
import jdbc.OpRegistre;
import jdbc.OperacionsBD;
import org.apache.commons.fileupload.FileItem;
import org.apache.commons.fileupload.disk.DiskFileItemFactory;
import org.apache.commons.fileupload.servlet.ServletFileUpload;

/**
 *
 * @author Carles Torras Font
 */
@WebServlet("/GestorNormes")
@MultipartConfig
public class GestorNormes extends HttpServlet {
    private static final long serialVersionUID = 1L;

    // location to store file uploaded
    public static String UPLOAD_DIRECTORY = "arxius" + File.separator + "documents";

    // upload settings
    private static final int MEMORY_THRESHOLD = 1024 * 1024 * 3; // 3MB
    private static final int MAX_FILE_SIZE = 1024 * 1024 * 40; // 40MB
    private static final int MAX_REQUEST_SIZE = 1024 * 1024 * 50; // 50MB
    public static String fileName;

    /**
     * Processes requests for both HTTP <code>GET</code> and <code>POST</code>
     * methods.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    protected void processRequest(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        response.setContentType("text/html;charset=UTF-8");
        request.setCharacterEncoding("UTF-8");
    }

    /**
     * Handles the HTTP <code>POST</code> method.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    @Override
    protected void doPost(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {

        // Configuro dades per desar arxius. Uso lib Apache Upload ----->
        DiskFileItemFactory factory = new DiskFileItemFactory();
        // sets memory threshold - beyond which files are stored in disk
        factory.setSizeThreshold(MEMORY_THRESHOLD);
        // sets temporary location to store files
        factory.setRepository(new File(System.getProperty("java.io.tmpdir")));
        ServletFileUpload upload = new ServletFileUpload(factory);
        // sets maximum size of upload file
        upload.setFileSizeMax(MAX_FILE_SIZE);
        // sets maximum size of request (include file + form data)
        upload.setSizeMax(MAX_REQUEST_SIZE);
        // constructs the directory path to store upload file
        // this path is relative to application's directory
        String uploadPath = getServletContext().getRealPath("") + File.separator + UPLOAD_DIRECTORY;
        // creates the directory if it does not exist
        File uploadDir = new File(uploadPath);
        if (!uploadDir.exists()) {
            uploadDir.mkdir();
        }
        // ----->

        // ----- Carrego la info del formulari ----->
        String accio = null;
        try {
            // parses the request's content to extract file data
            @SuppressWarnings("unchecked")
            //List<FileItem> formItems = new ArrayList<FileItem>();
            List<FileItem> formItems = upload.parseRequest(request);
            String arxiu = null;
            String titol = null;
            String id = null;
            String lknorma = null;
            //System.out.println("Control0");

            if (formItems != null && formItems.size() > 0) {
                // iterates over form's fields
                //System.out.println("Control01");
                for (FileItem item : formItems) {
                    // processes only fields that are not form fields
                    //System.out.println(item.getFieldName());
                    if (!item.isFormField()) {
                        //Agafo l'arxiu
                    }
                }
            }
        }
    }
}

```



```

import java.util.ArrayList;
import java.util.List;
import java.util.logging.Level;
import java.util.logging.Logger;
import javax.servlet.ServletException;
import javax.servlet.annotation.MultipartConfig;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;
import jdbc.OpRegistre;
import jdbc.OperacionsBD;
import org.apache.commons.fileupload.FileItem;
import org.apache.commons.fileupload.disk.DiskFileItemFactory;
import org.apache.commons.fileupload.servlet.ServletFileUpload;

/**
 *
 * @author Carles Torras Font
 */
@WebServlet("/GestorProfessionals")
@MultipartConfig
public class GestorProfessionals extends HttpServlet {
    private static final long serialVersionUID = 1L;

    // location to store file uploaded
    public static String UPLOAD_DIRECTORY = "arxius" + File.separator + "documents";

    // upload settings
    private static final int MEMORY_THRESHOLD = 1024 * 1024 * 3; // 3MB
    private static final int MAX_FILE_SIZE = 1024 * 1024 * 40; // 40MB
    private static final int MAX_REQUEST_SIZE = 1024 * 1024 * 50; // 50MB
    public static String fileName;

    /**
     * Processes requests for both HTTP <code>GET</code> and <code>POST</code>
     * methods.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    protected void processRequest(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        response.setContentType("text/html;charset=UTF-8");
        request.setCharacterEncoding("UTF-8");
    }

    /**
     * Handles the HTTP <code>POST</code> method.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    @Override
    protected void doPost(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {

        // Configuro dades per desar arxius. Uso lib Apache Upload ----->
        DiskFileItemFactory factory = new DiskFileItemFactory();
        // sets memory threshold - beyond which files are stored in disk
        factory.setSizeThreshold(MEMORY_THRESHOLD);
        // sets temporary location to store files
        factory.setRepository(new File(System.getProperty("java.io.tmpdir")));
        ServletFileUpload upload = new ServletFileUpload(factory);
        // sets maximum size of upload file
        upload.setFileSizeMax(MAX_FILE_SIZE);
        // sets maximum size of request (include file + form data)
        upload.setSizeMax(MAX_REQUEST_SIZE);
        // constructs the directory path to store upload file
        // this path is relative to application's directory
        String uploadPath = getServletContext().getRealPath("/") + File.separator + UPLOAD_DIRECTORY;
        // creates the directory if it does not exist
        File uploadDir = new File(uploadPath);
        if (!uploadDir.exists()) {
            uploadDir.mkdir();
        }
        // ----->

        // -----> Carrego la info del formulari ----->
        String accio = null;
        try {
            // parses the request's content to extract file data
            @SuppressWarnings("unchecked")
            //List<FileItem> formItems = new ArrayList<FileItem>();
            List<FileItem> formItems = upload.parseRequest(request);
            String arxiu = null;
            String nom = null;
            String cognom1 = null;
            String cognom2 = null;
            String nif = null;
            String nifg = null;
            String categoria = null;
            String password = null;
            String tutor = null;

```

```

String horariprof = null;
String actiu = "no";
//System.out.println("Control0");

if (formItems != null && formItems.size() > 0) {
    // iterates over form's fields
    //System.out.println("Control01");
    for (FormItem item : formItems) {
        // processes only fields that are not form fields
        //System.out.println(item.getFieldName());
        if (!item.isFormField()) {
            //Agafo l'arxiu
            fileName = new File(item.getName().getName());
            int mida = fileName.length();
            if (mida > 0) {
                fileName = fileName.replaceAll(" ", "_");
                storeFile = new File(getServletContext().getRealPath("") +
                    File.separator + UPLOAD_DIRECTORY + File.separator + "horariprof" + File.separator +
                    fileName);
                item.write(storeFile); // deso l'arxiu original
                arxiu = UPLOAD_DIRECTORY + File.separator + "horariprof" + File.separator + fileName;
                arxiu = arxiu.replaceAll("\\\\", "/");
            }
        } else {
            // Deso les informacions
            if ("nom".equals(item.getFieldName())) {
                nom = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
            } else if ("cognom1".equals(item.getFieldName())) {
                cognom1 = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
            } else if ("cognom2".equals(item.getFieldName())) {
                cognom2 = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
            } else if ("nifa".equals(item.getFieldName())) {
                nif = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
            } else if ("nifg".equals(item.getFieldName())) {
                nifg = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
                nifg = nifg.substring(Math.max(nifg.length() - 9, 0));
            } else if ("categoria".equals(item.getFieldName())) {
                categoria = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
            } else if ("actiu".equals(item.getFieldName())) {
                actiu = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
            }
            // Aquest següent només s'usa quan modifiquem arxiu:
            } else if ("horariprof".equals(item.getFieldName())) {
                horariprof = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
            } else if ("optionsRadios1".equals(item.getFieldName())) {
                tutor = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
            } else if ("password".equals(item.getFieldName())) {
                password = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
                // Xifro els passwords
                StringBuffer hexString = new StringBuffer();
                try {
                    MessageDigest md = MessageDigest.getInstance("SHA-256");
                    md.update(password.getBytes());
                    byte byteData[] = md.digest();
                    for (int i=0;i<byteData.length;i++) {
                        String hex=Integer.toHexString(0xff & byteData[i]);
                        if(hex.length()==1) hexString.append('0');
                        hexString.append(hex);
                    }
                } catch (NoSuchAlgorithmException ex) {
                    Logger.getLogger(GestorProfessionals.class.getName()).log(Level.SEVERE, null, ex);
                }
                password = hexString.toString();
            } else if ("optionsRadios2".equals(item.getFieldName())) {
                accio = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
                //System.out.println("Control1");
            }
        }
    }
}

// Recupero nom d'usuari de la sessió
HttpSession session = request.getSession();
String usuari_sessio = (String)session.getAttribute("usuari");

// Si afegeixo, tindrà un arxiu no null, si modifico, si tinc el null em quedo amb
// el que tenia, i si no tinc null, em quedo amb el nou (que llavors ja serà arxiu
if (arxiu==null) arxiu = horariprof;

if ("afegir".equals(accio)) {
    OperacionsBD op = new OperacionsBD();
    int registre=op.crearProfessional(new Professional(nif, tutor, arxiu, nom, cognom1, cognom2, password,
categoria, null, actiu));
    request.setAttribute("informacio", "Professional inserit correctament. Gràcies.");
    System.out.println(registre);
    // Deso registre
    List<String> nifs = new ArrayList<>();
    nifs.add(nif);
    OpRegistre or = new OpRegistre(); int i = or.nouRegistre("afegir_professional",usuari_sessio,nifs);
    request.setAttribute("resultat", registre);
    getServletContext().getRequestDispatcher("/WEB-INF/gestor/preindex.jsp").forward(request, response);
} else if ("modificar".equals(accio)) {
    OperacionsBD op = new OperacionsBD();
    Professional prof=op.cercaProfessional(nifg);
    request.setAttribute("professional", prof);
    // Llista de categories
    List<String> categories = new ArrayList<String>();
    OperacionsBD op3 = new OperacionsBD();
    categories = op3.llistarCategories();
    request.setAttribute("categories", categories);
    getServletContext().getRequestDispatcher("/WEB-INF/gestor/Modificarp.jsp").forward(request, response);
}

```

```

    } else if ("modificarex".equals(accio)) {
        OperacionsBD op = new OperacionsBD();
        int registre=op.modificarProfessional(nif, tutor, arxiu, nom, cognom1, cognom2, categoria, actiu);
        request.setAttribute("informacio", "Professional modificat correctament. Gràcies.");
        // Deso registre
        List <String> nifs = new ArrayList<>();
        nifs.add(nif);
        OpRegistre or = new OpRegistre(); int i = or.nouRegistre("modificar_professional",usuari_sessio,nifs);
        request.setAttribute("resultat", registre);
        getServletContext().getRequestDispatcher("/WEB-INF/gestor/preindex.jsp").forward(request, response);
    } else if ("eliminar".equals(accio)) {
        OperacionsBD op = new OperacionsBD();
        int registre=op.eliminarProfessional(nifg);
        request.setAttribute("informacio", "Professional eliminat correctament. Gràcies.");
        // Deso registre
        OpRegistre or = new OpRegistre(); int i = or.nouRegistre("eliminar_professional",usuari_sessio,null);
        request.setAttribute("resultat", registre);
        getServletContext().getRequestDispatcher("/WEB-INF/gestor/preindex.jsp").forward(request, response);
    } else if ("llistar".equals(accio)) {
        OperacionsBD op = new OperacionsBD();
        List<Professional> llistap = op.cercaProf();
        request.setAttribute("llistap", llistap);
        // Deso registre
        List<String> nifs4 = new ArrayList<>();
        for (int i = 0; i < llistap.size(); i++) {
            String elnifdara = llistap.get(i).getNif();
            nifs4.add(elnifdara);
        }
        OpRegistre or = new OpRegistre(); int i = or.nouRegistre("llistar_professional",usuari_sessio,nifs4);
        getServletContext().getRequestDispatcher("/WEB-INF/gestor/Llistap.jsp").forward(request, response);
    }
} catch (Exception ex) {
    request.setAttribute("message",
        "Hi ha hagut un error: " + ex.getMessage());
}
}
}
}

```

GestorReunions.java

```

/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package servlets;

import beans.Profreunions;
import beans.Reunio;
import java.io.File;
import java.io.IOException;
import java.util.ArrayList;
import java.util.List;
import javax.servlet.ServletException;
import javax.servlet.annotation.MultipartConfig;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;
import jdbc.OpRegistre;
import jdbc.OperacionsBD;
import org.apache.commons.fileupload.FileItem;
import org.apache.commons.fileupload.disk.DiskFileItemFactory;
import org.apache.commons.fileupload.servlet.ServletFileUpload;

/**
 *
 * @author Carles Torras Font
 */
@WebServlet("/GestorReunions")
@MultipartConfig
public class GestorReunions extends HttpServlet {
    private static final long serialVersionUID = 1L;

    // location to store file uploaded
    public static String UPLOAD_DIRECTORY = "arxius" + File.separator + "documents";

    // upload settings
    private static final int MEMORY_THRESHOLD = 1024 * 1024 * 3; // 3MB
    private static final int MAX_FILE_SIZE = 1024 * 1024 * 40; // 40MB
    private static final int MAX_REQUEST_SIZE = 1024 * 1024 * 50; // 50MB
    public static String fileName;

    /**
     * Processes requests for both HTTP <code>GET</code> and <code>POST</code>
     * methods.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    protected void processRequest(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        response.setContentType("text/html;charset=UTF-8");
    }
}

```

```

    request.setCharacterEncoding("UTF-8");
}

/**
 * Handles the HTTP <code>POST</code> method.
 *
 * @param request servlet request
 * @param response servlet response
 * @throws ServletException if a servlet-specific error occurs
 * @throws IOException if an I/O error occurs
 */
@Override
protected void doPost(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {

    // Configuro dades per desar arxius. Uso lib Apache Upload ----->
    DiskFileItemFactory factory = new DiskFileItemFactory();
    // sets memory threshold - beyond which files are stored in disk
    factory.setSizeThreshold(MEMORY_THRESHOLD);
    // sets temporary location to store files
    factory.setRepository(new File(System.getProperty("java.io.tmpdir")));
    ServletFileUpload upload = new ServletFileUpload(factory);
    // sets maximum size of upload file
    upload.setFileSizeMax(MAX_FILE_SIZE);
    // sets maximum size of request (include file + form data)
    upload.setSizeMax(MAX_REQUEST_SIZE);
    // constructs the directory path to store upload file
    // this path is relative to application's directory
    String uploadPath = getServletContext().getRealPath("") + File.separator + UPLOAD_DIRECTORY;
    // creates the directory if it does not exist
    File uploadDir = new File(uploadPath);
    if (!uploadDir.exists()) {
        uploadDir.mkdir();
    }
    // ----->
    // -----> Carrego la info del formulari ----->
    String accio = null;
    try {
        // parses the request's content to extract file data
        @SuppressWarnings("unchecked")
        //List<FileItem> formItems = new ArrayList<FileItem>();
        List<FileItem> formItems = upload.parseRequest(request);
        String assistents = null;
        String datahora = null;
        String arxiu = null;
        String lkcontingut = null;
        String id = null;
        String propera = null;
        String alumne = null;
        String hihaanterior = null;
        List<String> professionals = new ArrayList<>();

        if (formItems != null && formItems.size() > 0) {
            // iterates over form's fields
            //System.out.println("Control01");
            for (FileItem item : formItems) {
                // processes only fields that are not form fields
                //System.out.println(item.getFieldName());
                if (!item.isFormField()) {
                    //Agafo l'arxiu
                    fileName = new File(item.getName()).getName();
                    int mida = fileName.length();
                    if (mida > 0) {
                        fileName = fileName.replaceAll(" ", "_");
                        File storeFile = new File(getServletContext().getRealPath("") +
                            File.separator + UPLOAD_DIRECTORY + File.separator + "reunions" + File.separator +
                                fileName);
                        item.write(storeFile); // deso l'arxiu original
                        arxiu = UPLOAD_DIRECTORY + File.separator + "reunions" + File.separator + fileName;
                        arxiu = arxiu.replaceAll("\\\\", "////////");
                    }
                } else {
                    // Deso les informacions
                    if ("assistents".equals(item.getFieldName())) {
                        assistents = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
                    } else if ("idanterior".equals(item.getFieldName())) {
                        propera = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
                    } else if ("optionsRadios2".equals(item.getFieldName())) {
                        accio = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
                    } else if ("optionsRadios3".equals(item.getFieldName())) {
                        hihaanterior = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
                    } else if ("datahora".equals(item.getFieldName())) {
                        datahora = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
                        //System.out.println("datahora: "+datahora);
                    } else if ("id".equals(item.getFieldName())) {
                        id = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
                    } // Aquest següent només s'usa quan modifiquem arxiu:
                    } else if ("lkcontingut".equals(item.getFieldName())) {
                        lkcontingut = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
                    } else if ("alumnes".equals(item.getFieldName())) {
                        alumne = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
                        alumne = alumne.substring(Math.max(alumne.length() - 9, 0));
                    } else if ("professionals".equals(item.getFieldName())) {
                        String loprofessional = new String(item.getString().getBytes("ISO-8859-1"), "UTF-8");
                        loprofessional = loprofessional.substring(Math.max(loprofesional.length() - 9, 0));
                        professionals.add(loprofesional);
                    }
                }
            }
        }
    }
}

```

```

    }
}

// Definició de reunió següent
if ("nonhiha".equals(hihaanterior)) propera=null;

// Recupero nom d'usuari
HttpSession session = request.getSession();
String professional = (String)session.getAttribute("usuari");
String rol = (String)session.getAttribute("rol");

// Si afegeixo, tindrà un arxIU no null, si modifico, si tinc el null em quedo amb
// el que tenia, i si no tinc null, em quedo amb el nou (que llavors ja serà arxIU
if (arxiu==null) arxiu = lkcontingut;

//System.out.println("Acció: "+accio);
if ("afegir".equals(accio)) {
    // Creo l'informe
    // Primer, asseguro que jo hi sóc...
    int j = 0;
    for (int i=0; i<professionals.size(); i++) {
        if (professionals.get(i).equals(professional)) j = 1;
    }
    if (j==0) professionals.add(professional);
    OperacionsBD op = new OperacionsBD();
    int registre=op.crearReunio(datahora, assistents, arxiu, propera, alumne);
    // I omplio la taula que afegeix professionals a l'informe
    OperacionsBD op2 = new OperacionsBD();
    int registre2=op2.crearProfReunions(String.valueOf(registre), professionals);
    request.setAttribute("informacio", "Reunió inserida correctament. Gràcies.");
    // Deso registre
    List<String> nifs = new ArrayList<>();
    nifs.add(alumne);
    OpRegistre or = new OpRegistre(); int i = or.nouRegistre("afegir_reunions",professional,nifs);
    request.setAttribute("resultat", registre2);
    getServletContext().getRequestDispatcher("/WEB-INF/gestor/preindex.jsp").forward(request, response);
} else if ("eliminar".equals(accio)) {
    // Elimino els professionals de l'informe
    OperacionsBD op2 = new OperacionsBD();
    int registre2=op2.eliminarProfReunions(id);
    // Elimino l'informe
    OperacionsBD op = new OperacionsBD();
    int registre=op.eliminarReunio(id);
    request.setAttribute("informacio", "Reunió eliminada correctament. Gràcies.");
    // Deso registre
    OpRegistre or = new OpRegistre(); int i = or.nouRegistre("eliminar_reunions",professional,null);
    request.setAttribute("resultat", registre);
    getServletContext().getRequestDispatcher("/WEB-INF/gestor/preindex.jsp").forward(request, response);
} else if ("llistar".equals(accio)) {
    // Agafo dades de la reunió
    OperacionsBD op = new OperacionsBD();
    List<Reunio> llistaa = new ArrayList<>();
    if ("Director/a".equals(rol) || "Administrador/a".equals(rol)) {
        llistaa = op.llistarReunions("%");
    } else {
        llistaa = op.llistarReunions(professional);
    }
    request.setAttribute("llistaa", llistaa);

    // Converteixo NIF en lo llegible (nom, cognom, nif)
    List<String> alumnes = new ArrayList<>();
    for (int i = 0; i < llistaa.size(); i++) {
        OperacionsBD opl = new OperacionsBD();
        String lotio = opl.cercaAlumne((llistaa.get(i).getNif()).toString());
        alumnes.add(lotio);
    }
    request.setAttribute("nomsalumnes", alumnes);

    // I dels professionals involucrats
    OperacionsBD op2 = new OperacionsBD();
    List<Profreunions> llistaal = op2.llistarProfReunions();
    request.setAttribute("llistaal", llistaal);

    // Converteixo NIF en lo llegible (nom, cognom, nif)
    List<String> profes = new ArrayList<>();
    for (int i = 0; i < llistaal.size(); i++) {
        OperacionsBD op3 = new OperacionsBD();
        String lotio = op3.cercaProfessional((llistaal.get(i).getNif()).toString());
        //System.out.println(lotio);
        profes.add(lotio);
    }
    request.setAttribute("nomsprofes", profes);

    // Llista d'alumnes del profe (només podrà modificar o eliminar el seus)
    List<String> nifs = new ArrayList<>();
    OperacionsBD op4 = new OperacionsBD();
    nifs = op4.NifAlumnesProf(professional);
    request.setAttribute("alumnesprofe", nifs);

    // Deso registre
    OpRegistre or = new OpRegistre(); int i = or.nouRegistre("llistar_reunions",professional,null);
    getServletContext().getRequestDispatcher("/WEB-INF/gestor/llistareu.jsp").forward(request, response);
} else if ("modificar".equals(accio)) {
    // Cerco l'informe
    OperacionsBD op = new OperacionsBD();
    Reunio reunio=op.cercaReunio(id);
    request.setAttribute("reunio", reunio);

    // Cerco els profes de la reunió (només NIF)

```



```

List<String> profesreunio = new ArrayList<>();
OperacionsBD op3 = new OperacionsBD();
profesreunio = op3.llistarProfReunions(id);
request.setAttribute("profesreunio", profesreunio);

// Carrego la llista d'alumnes del profe
// --- Primer la llista total
List<String> alumnes = new ArrayList<>();
OperacionsBD op5 = new OperacionsBD();
alumnes = op5.llistarTotsAlumnes();
// --- Després els nifs dels alumnes del profe
List<String> nifs = new ArrayList<>();
OperacionsBD op4 = new OperacionsBD();
nifs = op4.NifAlumnesProf(professional);
// --- I em quedo amb les interseccions
String lodara;
String lalumne = null;
List<String> alumnesprofe = new ArrayList<>();
for (int i=0; i<alumnes.size(); i++) {
    for (int j=0; j<nifs.size(); j++) {
        lodara = alumnes.get(i).substring(Math.max(alumnes.get(i).length() - 9, 0));
        if (lodara.equals(reunio.getNif())) {
            lalumne = alumnes.get(i);
        }
        if (lodara.equals(nifs.get(j))) alumnesprofe.add(alumnes.get(i));
    }
}
request.setAttribute("alumnesprofe", alumnesprofe);
request.setAttribute("lalumne", lalumne);

// Carrego la llista de professionals (Nom, NIF)
List<String> totslosprofessionals = new ArrayList<>();
List<String> totslosprofessionalsel = new ArrayList<>();
OperacionsBD op45 = new OperacionsBD();
totslosprofessionals = op45.llistarTotsProfs();
request.setAttribute("professionals", totslosprofessionals);
for (int i=0; i<totslosprofessionals.size(); i++) {
    lodara = totslosprofessionals.get(i).substring(Math.max(totslosprofessionals.get(i).length() - 9, 0));
    for (int j=0; j<profesreunio.size(); j++) {
        if (lodara.equals(profesreunio.get(j))) {
            totslosprofessionalsel.add(totslosprofessionals.get(i));
        }
    }
}
request.setAttribute("professionals1", totslosprofessionalsel);

//Recupero llista de reunions pel professional
List<String> reunions = new ArrayList<>();
OperacionsBD op26 = new OperacionsBD();
reunions = op26.llistarIdReunions(professional);
request.setAttribute("reunions", reunions);

getServletContext().getRequestDispatcher("/WEB-INF/gestor/ModificarReu.jsp").forward(request, response);
} else if ("modificarex".equals(accio)) {
    // Per informes faig un UPDATE
    // Però per profinformes, millor elimino i creo
    // Primer, elimino tot lo que hi ha de professionals...
    OperacionsBD op0 = new OperacionsBD();
    int registre1=op0.eliminarProfReunions(id);
    // Segon, asseguro que jo hi són...
    int j = 0;
    for (int i=0; i<professionals.size(); i++) {
        if (professionals.get(i).equals(professional)) j = 1;
    }
    if (j==0) professionals.add(professional);
    // Modifico l'informe
    OperacionsBD op = new OperacionsBD();
    int registre=op.modificarReunio(datahora, assistents, arxiu, propera, alumne, id);
    // I creo els professionals a l'informe
    OperacionsBD op2 = new OperacionsBD();
    int registre2=op2.crearProfReunions(id, professionals);
    request.setAttribute("informacio", "Reunió modificada correctament. Gràcies.");
    // Deso registre
    List <String> nifs = new ArrayList<>();
    nifs.add(alumne);
    OpRegistre or = new OpRegistre(); int i = or.nouRegistre("modificar_reunions",professional,nifs);
    request.setAttribute("resultat", registre);
    getServletContext().getRequestDispatcher("/WEB-INF/gestor/preindex.jsp").forward(request, response);
}
} catch (Exception ex) {
    request.setAttribute("message",
        "Hi ha hagut un error: " + ex.getMessage());
}
}
}
}

```

Index.java

```

/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

```

```
package servlets;
```

```

import java.io.IOException;
import java.security.MessageDigest;
import java.security.NoSuchAlgorithmException;
import java.util.ArrayList;
import java.util.List;
import java.util.logging.Level;
import java.util.logging.Logger;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;
import jdbc.OpRegistre;
import jdbc.OperacionsBD;

/**
 *
 * @author Dr.Carles Torras Font
 */
@WebServlet("/Index")
public class Index extends HttpServlet {

    /**
     * Processes requests for both HTTP <code>GET</code> and <code>POST</code>
     * methods.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    protected void processRequest(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        response.setContentType("text/html;charset=UTF-8");
    }

    // <editor-fold defaultstate="collapsed" desc="HttpServlet methods. Click on the + sign on the left to edit the code.">
    /**
     * Handles the HTTP <code>GET</code> method.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    @Override
    protected void doGet(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
    }

    /**
     * Handles the HTTP <code>POST</code> method.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    @Override
    protected void doPost(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {

        int dretaaccés = 0; // No tens dret a entrar

        // Comprovo si la sessió està ja iniciada
        HttpSession session = request.getSession();
        String usuari = (String)session.getAttribute("usuari");
        request.setAttribute("usuari", usuari);
        // Converteixo NIF en lo llegible (nom, cognom, nif)
        if (usuari != null) {
            OperacionsBD op00 = new OperacionsBD();
            String usuariltot = op00.cercaProfessional(usuari).toString();
            request.setAttribute("usuariltot", usuariltot);
            String fotousuari = op00.cercaProfessional(usuari).getLkfotousuari();
            request.setAttribute("fotousuari", fotousuari);
            dretaaccés = 1;
        }

        // Si no ho està :
        if (usuari == null) {
            // Llegeixo les dades de l'usuari
            String usuari = new String(request.getParameter("usuari").getBytes("ISO-8859-1"), "UTF-8");
            String clau = new String(request.getParameter("clau").getBytes("ISO-8859-1"), "UTF-8");
            // Xifro el password
            StringBuffer hexString = new StringBuffer();
            try {
                MessageDigest md = MessageDigest.getInstance("SHA-256");
                md.update(clau.getBytes());
                byte byteData[] = md.digest();
                for (int iii=0;iii<byteData.length;iii++) {
                    String hex=Integer.toHexString(0xff & byteData[iii]);
                    if(hex.length()==1) hexString.append("0");
                    hexString.append(hex);
                }
            } catch (NoSuchAlgorithmException ex) {
                Logger.getLogger(Index.class.getName()).log(Level.SEVERE, null, ex);
                //System.out.println("Error en encriptar la clau");
            }
        }
    }
}

```

```

    }
    clau = hexString.toString();

    // Valido l'usuari
    try {
        OperacionsBD op0 = new OperacionsBD();
        List<String> autenticacio = op0.Autenticacio(usuari);
        // Comprovo si l'autenticaciÃ³ Ã©s bona
        if (autenticacio.get(0).equals(clau)) {
            String rol = autenticacio.get(1);
            String actiu = autenticacio.get(2);

            if ("si".equals(actiu)) {
                // Declaro les 2 variables de sessiÃ³: usuari i rol
                synchronized(session) {
                    String roll = (String)session.getAttribute("rol");
                    usuariil = (String)session.getAttribute("usuari");
                    if (roll == null) roll = rol;
                    if (usuariil == null) usuariil = usuari;
                    session.setAttribute("rol", roll);
                    session.setAttribute("usuari", usuariil);
                }
                OperacionsBD op00 = new OperacionsBD();
                String usuariiltot = op00.cercaProfessional(usuariil).toString();
                String fotousuari = op00.cercaProfessional(usuariil).getLkfotousuari();
                request.setAttribute("usuariiltot", usuariiltot);
                request.setAttribute("fotousuari", fotousuari);

                // Deso l'accÃ©s al registre
                OpRegistre or = new OpRegistre();
                int i = or.nouRegistre("acces",usuariil,null);
                dretaacces = 1; // Validat, ja pots entrar!
                //System.out.println("Usuari validat!");
            } else {
                // Usuari no validat (no estÃ© actiu)
                request.setAttribute("resultat", 2);
                //System.out.println("Usuari no actiu");
            }
        } else {
            // Usuari no validat (password equipovat)
            request.setAttribute("resultat", 1);
            //System.out.println("Usuari no vÃ© lid");
        }
    } catch (Exception e) {
        // Per algun motiu, no he pogut validar l'usuari
        request.setAttribute("resultat", 3);
        //System.out.println("No s'ha pogut comprovar si l'usuari Ã©s vÃ© lid");
    }
}

// Si arribo aquÃ©, Ã©s que tinc la sessiÃ³ iniciada o que he validat l'usuari correctament.
// Llavors, procedeix amb l'accÃ©s:
if (dretaacces == 1) {

    //Recupero llista de professionals
    List<String> professionals = new ArrayList<>();
    OperacionsBD op45 = new OperacionsBD();
    professionals = op45.llistarTotsProfes();
    request.setAttribute("professionals", professionals);

    //Recupero llista d'alumnes
    List<String> alumnes = new ArrayList<>();
    OperacionsBD op5 = new OperacionsBD();
    alumnes = op5.llistarTotsAlumnes();
    request.setAttribute("alumnes", alumnes);

    //Recupero llista de municipis
    List<String> municipis = new ArrayList<>();
    OperacionsBD op = new OperacionsBD();
    municipis = op.llistarMunicipis();
    request.setAttribute("municipis", municipis);

    //Recupero llista d'aules
    List<String> aules = new ArrayList<>();
    OperacionsBD opl = new OperacionsBD();
    aules = opl.llistarClasseCurs();
    request.setAttribute("aules", aules);

    //Recupero llista de noms d'aules
    List<String> nomauls = new ArrayList<>();
    OperacionsBD opl1 = new OperacionsBD();
    nomauls = opl1.llistarNomAules();
    request.setAttribute("nomauls", nomauls);

    //Recupero llista de noms de cursos
    List<String> cursos = new ArrayList<>();
    OperacionsBD op6 = new OperacionsBD();
    cursos = op6.llistarCursos();
    request.setAttribute("cursos", cursos);

    //Recupero llista de categories
    List<String> categories = new ArrayList<>();
    OperacionsBD op3 = new OperacionsBD();
    categories = op3.llistarCategories();
    request.setAttribute("categories", categories);

    //Recupero llista d'alumnes del profe
    // -- NIF del profe

```

```

List<String> nifs = new ArrayList<>();
OperacionsBD op4 = new OperacionsBD();
nifs = op4.NifAlumnesProf(usuari1);
// -- Em quedo amb les interseccions
String lodara;
List<String> alumnesprofe = new ArrayList<>();
for (int i=0; i<alumnes.size(); i++) {
    for (int j=0; j<nifs.size(); j++) {
        lodara = alumnes.get(i).substring(Math.max(alumnes.get(i).length() - 9, 0));
        if (lodara.equals(nifs.get(j))) alumnesprofe.add(alumnes.get(i));
    }
}
request.setAttribute("alumnesprofe", alumnesprofe);

// Finalment, puc recollir resultats d'operacions fetes que passen per aquà-
try {
    int resultat = Integer.parseInt(request.getParameter("resultat"));
    request.setAttribute("resultat", resultat);
} catch (NumberFormatException e) {}

// I a l'ataque!!
getServletContext().getRequestDispatcher("/WEB-INF/gestor/index.jsp").forward(request, response);
} else {
    getServletContext().getRequestDispatcher("/index.jsp").forward(request, response);
}
}
}
}

```

TancarSessio.java

```

/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

package servlets;

import java.io.IOException;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;

/**
 *
 * @author Dr.Carles Torras Font
 */
@WebServlet("/TancarSessio")
public class TancarSessio extends HttpServlet {

    /**
     * Processes requests for both HTTP <code>GET</code> and <code>POST</code>
     * methods.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    protected void processRequest(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        response.setContentType("text/html;charset=UTF-8");
    }

    // <editor-fold defaultstate="collapsed" desc="HttpServlet methods. Click on the + sign on the left to edit the code.">
    /**
     * Handles the HTTP <code>GET</code> method.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    @Override
    protected void doGet(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
    }

    /**
     * Handles the HTTP <code>POST</code> method.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    @Override
    protected void doPost(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {

        // Comprovo si la sessió està ja iniciada
        HttpSession session = request.getSession();
    }
}

```

```

        session.invalidate();
        getServletContext().getRequestDispatcher("/index.jsp").forward(request, response);
    }
}

```

TutIAI.java

```

/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package servlets;

import java.io.IOException;
import java.util.ArrayList;
import java.util.List;
import java.util.logging.Level;
import java.util.logging.Logger;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import jdbc.OperacionsBD;
import org.json.JSONException;
import org.json.JSONObject;

/**
 *
 * @author Carles Torras Font
 */
@WebServlet(name = "TutIAI", urlPatterns = {"/TutIAI"})
public class TutIAI extends HttpServlet {

    /**
     * Processes requests for both HTTP GET and POST
     * methods.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    protected void processRequest(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        response.setContentType("text/html;charset=UTF-8");
    }

    /**
     * Handles the HTTP POST method.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    @Override
    protected void doPost(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {

        String aula_nom = request.getParameter("nomaula");
        String aula_curs = request.getParameter("curs");

        JSONObject temporal = new JSONObject();

        // ALUMNES -----
        // Llistat d'alumnes total
        OperacionsBD op = new OperacionsBD();
        List<String> llistaalumnes = op.llistarTotsAlumnes();
        // Llistat actual de nif's d'alumnes de la classe
        OperacionsBD op2 = new OperacionsBD();
        List<String> nifsalumnesaula = op2.llistarAlumnesClasse(aula_nom, aula_curs);
        // De la llistat de tots els alumnes, pillo els que NIF coincideixi
        String partnif;
        List<String> alumnesaula = new ArrayList<>();
        for (int i = 0; i < llistaalumnes.size(); i++) {
            partnif = llistaalumnes.get(i).substring(Math.max(llistaalumnes.get(i).length() - 9, 0));
            for (int j = 0; j < nifsalumnesaula.size(); j++) {
                if (partnif.equals(nifsalumnesaula.get(j))) {
                    try {
                        temporal.append("alumnesaula", llistaalumnes.get(i));
                    } catch (JSONException ex) {
                        Logger.getLogger(TutIAI.class.getName()).log(Level.SEVERE, null, ex);
                    }
                }
            }
        }

        // PROFES -----
        // Llistat de profes total
        OperacionsBD op11 = new OperacionsBD();
        List<String> llistatutors = op11.llistarTutors();
        // Llistat actual de nif's de tutors de la classe
        OperacionsBD op12 = new OperacionsBD();
        List<String> nifsprefesaula = op12.llistarProfesClasse(aula_nom, aula_curs);
    }
}

```

```
// De la llistat de tots els tutors, pillo els que NIF coincideixi
String partnif2;
List<String> profesaula = new ArrayList<>();
for (int i = 0; i < llistatutors.size(); i++) {
    partnif2 = llistatutors.get(i).substring(Math.max(llistatutors.get(i).length() - 9, 0));
    for (int j = 0; j < nifsprofesaula.size(); j++) {
        if (partnif2.equals(nifsprofesaula.get(j))) {
            try {
                temporal.append("profesaula", llistatutors.get(i));
            } catch (JSONException ex) {
                Logger.getLogger(TutIAl.class.getName()).log(Level.SEVERE, null, ex);
            }
        }
    }
}

//System.out.println("sortida: "+temporal);
response.setContentType("application/json");
response.getWriter().write(temporal.toString());
processRequest(request, response);
}
```

A4.3.- Gestor de còpies de seguretat

Paquet copiabd

CopiaBD.java
CopiaBDIG.form
CopiaBDIG.java
Executa.java

CopiaBD.java

```

/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

package copiabd;

/**
 * Programa de backup de les dades de la BD i arxius associats del Jeroni
 * -----
 *
 * El programa realitza diàriament una còpia de les dades: (1) es crea un
 * backup de la BD en un arxIU SQL, (2) es crea un ZIP on s'hi desen tots
 * els arxius associats ubicats en la carpeta del Glassfish i també s'hi
 * afegeix l'arxIU SQL anteriorment creat. Tot queda desat, doncs, en el
 * ZIP.
 * La còpia diària (ZIP amb el nom especificat) es sobre-escriu cada dia.
 * PerÀ addicionalment, 1 cop cada setmana es desa també un arxIU que no
 * es sobre-escriu amb la mateixa informaciÓ. Aquest arxIU es titula
 * nomarxiu_DDMMAA_HHMMSS.ZIP
 *
 * @author Dr. Carles Torras Font
 */
public class CopiaBD {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        /* Set the Nimbus look and feel */
        //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">
        /* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.
         * For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
         */
        try {
            for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {
                if ("Windows".equals(info.getName())) {
                    javax.swing.UIManager.setLookAndFeel(info.getClassName());
                    break;
                }
            }
        } catch (ClassNotFoundException ex) {
            java.util.logging.Logger.getLogger(CopiaBDIG.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
        } catch (InstantiationException ex) {
            java.util.logging.Logger.getLogger(CopiaBDIG.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
        } catch (IllegalAccessException ex) {
            java.util.logging.Logger.getLogger(CopiaBDIG.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
        } catch (javax.swing.UnsupportedLookAndFeelException ex) {
            java.util.logging.Logger.getLogger(CopiaBDIG.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
        }
        //</editor-fold>

        /* Create and display the form */
        java.awt.EventQueue.invokeLater(new Runnable() {
            public void run() {
                new CopiaBDIG().setVisible(true);
            }
        });
    }
}

```

CopiaBDIG.form

```

<?xml version="1.0" encoding="UTF-8" ?>
<Form version="1.3" maxVersion="1.9" type="org.netbeans.modules.form.forminfo.JFrameFormInfo">
  <Properties>
    <Property name="defaultCloseOperation" type="int" value="3"/>
    <Property name="minimumSize" type="java.awt.Dimension" editor="org.netbeans.beaninfo.editors.DimensionEditor">
      <Dimension value="[484, 280]"/>
    </Property>
    <Property name="preferredSize" type="java.awt.Dimension" editor="org.netbeans.beaninfo.editors.DimensionEditor">
      <Dimension value="[484, 280]"/>
    </Property>
  </Properties>
</Form>

```

```

</Properties>
<SyntheticProperties>
  <SyntheticProperty name="formSizePolicy" type="int" value="1"/>
  <SyntheticProperty name="generateCenter" type="boolean" value="false"/>
</SyntheticProperties>
<AuxValues>
  <AuxValue name="FormSettings_autoResourcing" type="java.lang.Integer" value="0"/>
  <AuxValue name="FormSettings_autoSetComponentName" type="java.lang.Boolean" value="false"/>
  <AuxValue name="FormSettings_generateFQN" type="java.lang.Boolean" value="true"/>
  <AuxValue name="FormSettings_generateMnemonicsCode" type="java.lang.Boolean" value="false"/>
  <AuxValue name="FormSettings_i18nAutoMode" type="java.lang.Boolean" value="false"/>
  <AuxValue name="FormSettings_layoutCodeTarget" type="java.lang.Integer" value="1"/>
  <AuxValue name="FormSettings_listenerGenerationStyle" type="java.lang.Integer" value="0"/>
  <AuxValue name="FormSettings_variablesLocal" type="java.lang.Boolean" value="false"/>
  <AuxValue name="FormSettings_variablesModifier" type="java.lang.Integer" value="2"/>
  <AuxValue name="designerSize" type="java.awt.Dimension" value="-84,-
19,0,5,115,114,0,18,106,97,118,97,46,97,119,116,46,68,105,109,101,110,115,105,111,110,65,-114,-39,-41,-
84,95,68,20,2,0,2,73,0,6,104,101,105,103,104,116,73,0,5,119,105,100,116,104,120,112,0,0,0,-28,0,0,1,-37"/>
</AuxValues>

<Layout class="org.netbeans.modules.form.compat2.layouts.DesignAbsoluteLayout">
  <Property name="useNullLayout" type="boolean" value="false"/>
</Layout>
<SubComponents>
  <Component class="javax.swing.JLabel" name="jLabel1">
    <Properties>
      <Property name="text" type="java.lang.String" value="by Carles Torras, 2015"/>
    </Properties>
    <Constraints>
      <Constraint layoutClass="org.netbeans.modules.form.compat2.layouts.DesignAbsoluteLayout"
value="org.netbeans.modules.form.compat2.layouts.DesignAbsoluteLayout$AbsoluteConstraintsDescription">
        <AbsoluteConstraints x="30" y="30" width="-1" height="-1"/>
      </Constraint>
    </Constraints>
  </Component>
  <Component class="javax.swing.JLabel" name="jLabel2">
    <Properties>
      <Property name="font" type="java.awt.Font" editor="org.netbeans.beaninfo.editors.FontEditor">
        <Font name="Tahoma" size="14" style="1"/>
      </Property>
      <Property name="text" type="java.lang.String" value="Benvingut al gestor d'execució de còpies de seguretat per MySQL"/>
    </Properties>
    <Constraints>
      <Constraint layoutClass="org.netbeans.modules.form.compat2.layouts.DesignAbsoluteLayout"
value="org.netbeans.modules.form.compat2.layouts.DesignAbsoluteLayout$AbsoluteConstraintsDescription">
        <AbsoluteConstraints x="10" y="10" width="-1" height="-1"/>
      </Constraint>
    </Constraints>
  </Component>
  <Container class="javax.swing.JPanel" name="jPanel1">
    <Constraints>
      <Constraint layoutClass="org.netbeans.modules.form.compat2.layouts.DesignAbsoluteLayout"
value="org.netbeans.modules.form.compat2.layouts.DesignAbsoluteLayout$AbsoluteConstraintsDescription">
        <AbsoluteConstraints x="40" y="50" width="-1" height="120"/>
      </Constraint>
    </Constraints>

    <Layout>
      <DimensionLayout dim="0">
        <Group type="103" groupAlignment="0" attributes="0">
          <Group type="102" alignment="1" attributes="0">
            <EmptySpace max="32767" attributes="0"/>
            <Group type="103" groupAlignment="0" attributes="0">
              <Component id="jLabel4" alignment="1" min="-2" max="-2" attributes="0"/>
              <Component id="jLabel3" alignment="1" min="-2" max="-2" attributes="0"/>
              <Component id="jLabel5" alignment="1" min="-2" max="-2" attributes="0"/>
              <Component id="jLabel6" alignment="1" min="-2" max="-2" attributes="0"/>
            </Group>
            <EmptySpace type="unrelated" max="-2" attributes="0"/>
            <Group type="103" groupAlignment="0" attributes="0">
              <Group type="103" groupAlignment="0" max="-2" attributes="0">
                <Component id="arxiu" max="32767" attributes="0"/>
                <Component id="basedades" max="32767" attributes="0"/>
                <Component id="clau" pref="120" max="32767" attributes="0"/>
              </Group>
              <Component id="disc" min="-2" pref="36" max="-2" attributes="0"/>
            </Group>
            <EmptySpace max="-2" attributes="0"/>
          </Group>
        </Group>
      </DimensionLayout>
      <DimensionLayout dim="1">
        <Group type="103" groupAlignment="0" attributes="0">
          <Group type="102" alignment="0" attributes="0">
            <EmptySpace max="-2" attributes="0"/>
            <Group type="103" groupAlignment="3" attributes="0">
              <Component id="clau" alignment="3" min="-2" max="-2" attributes="0"/>
              <Component id="jLabel4" alignment="3" min="-2" max="-2" attributes="0"/>
            </Group>
            <EmptySpace max="-2" attributes="0"/>
            <Group type="103" groupAlignment="3" attributes="0">
              <Component id="basedades" alignment="3" min="-2" max="-2" attributes="0"/>
              <Component id="jLabel3" alignment="3" min="-2" max="-2" attributes="0"/>
            </Group>
            <EmptySpace max="-2" attributes="0"/>
            <Group type="103" groupAlignment="3" attributes="0">
              <Component id="arxiu" alignment="3" min="-2" max="-2" attributes="0"/>
              <Component id="jLabel5" alignment="3" min="-2" max="-2" attributes="0"/>
            </Group>
          </Group>
        </Group>
      </DimensionLayout>
    </Layout>
  </Container>
</SubComponents>

```



```

        <EmptySpace max="-2" attributes="0"/>
        <Group type="103" groupAlignment="3" attributes="0">
            <Component id="disc" alignment="3" min="-2" max="-2" attributes="0"/>
            <Component id="jLabel6" alignment="3" min="-2" max="-2" attributes="0"/>
        </Group>
        <EmptySpace max="32767" attributes="0"/>
    </Group>
</Group>
</DimensionLayout>
</Layout>
<SubComponents>
    <Component class="javax.swing.JTextField" name="basedades">
        <Properties>
            <Property name="text" type="java.lang.String" value="jeroni"/>
        </Properties>
        <Events>
            <EventHandler
                event="actionPerformed"
                listener="java.awt.event.ActionListener"
                parameters="java.awt.event.ActionEvent" handler="basedadesActionPerformed"/>
        </Events>
    </Component>
    <Component class="javax.swing.JLabel" name="jLabel3">
        <Properties>
            <Property name="text" type="java.lang.String" value="Base de dades:"/>
        </Properties>
    </Component>
    <Component class="javax.swing.JPasswordField" name="clau">
        <Properties>
            <Property name="text" type="java.lang.String" value="password"/>
        </Properties>
        <Events>
            <EventHandler
                event="actionPerformed"
                listener="java.awt.event.ActionListener"
                parameters="java.awt.event.ActionEvent" handler="clauActionPerformed"/>
        </Events>
    </Component>
    <Component class="javax.swing.JLabel" name="jLabel4">
        <Properties>
            <Property name="text" type="java.lang.String" value="Password del root:"/>
        </Properties>
    </Component>
    <Component class="javax.swing.JTextField" name="arxiu">
        <Properties>
            <Property name="text" type="java.lang.String" value="jeronisql"/>
        </Properties>
        <Events>
            <EventHandler
                event="actionPerformed"
                listener="java.awt.event.ActionListener"
                parameters="java.awt.event.ActionEvent" handler="arxiuActionPerformed"/>
        </Events>
    </Component>
    <Component class="javax.swing.JLabel" name="jLabel5">
        <Properties>
            <Property name="text" type="java.lang.String" value="Arxiu:"/>
        </Properties>
    </Component>
    <Component class="javax.swing.JTextField" name="disc">
        <Properties>
            <Property name="text" type="java.lang.String" value="e"/>
        </Properties>
        <Events>
            <EventHandler
                event="actionPerformed"
                listener="java.awt.event.ActionListener"
                parameters="java.awt.event.ActionEvent" handler="discActionPerformed"/>
        </Events>
    </Component>
    <Component class="javax.swing.JLabel" name="jLabel6">
        <Properties>
            <Property name="text" type="java.lang.String" value="Unitat de disc (sortida):"/>
        </Properties>
    </Component>
</SubComponents>
</Container>
<Component class="javax.swing.JButton" name="Sortir">
    <Properties>
        <Property name="text" type="java.lang.String" value="Sortir"/>
    </Properties>
    <Events>
        <EventHandler event="actionPerformed" listener="java.awt.event.ActionListener" parameters="java.awt.event.ActionEvent"
            handler="SortirActionPerformed"/>
    </Events>
    <Constraints>
        <Constraint
            layoutClass="org.netbeans.modules.form.compat2.layouts.DesignAbsoluteLayout"
            value="org.netbeans.modules.form.compat2.layouts.DesignAbsoluteLayout$AbsoluteConstraintsDescription">
            <AbsoluteConstraints x="20" y="200" width="440" height="-1"/>
        </Constraint>
    </Constraints>
</Component>
<Component class="javax.swing.JButton" name="Manual">
    <Properties>
        <Property name="text" type="java.lang.String" value="Manual"/>
    </Properties>
    <Events>
        <EventHandler event="actionPerformed" listener="java.awt.event.ActionListener" parameters="java.awt.event.ActionEvent"
            handler="ManualActionPerformed"/>
    </Events>
    <Constraints>
        <Constraint
            layoutClass="org.netbeans.modules.form.compat2.layouts.DesignAbsoluteLayout"
            value="org.netbeans.modules.form.compat2.layouts.DesignAbsoluteLayout$AbsoluteConstraintsDescription">
            <AbsoluteConstraints x="20" y="170" width="230" height="-1"/>
        </Constraint>
    </Constraints>
</Component>

```

```

<Component class="javax.swing.JTextField" name="temps">
  <Properties>
    <Property name="text" type="java.lang.String" value="24"/>
  </Properties>
  </Events>
  <EventHandler event="actionPerformed" listener="java.awt.event.ActionListener" parameters="java.awt.event.ActionEvent"
handler="tempsActionPerformed"/>
  </Events>
  <Constraints>
    <Constraint
value="org.netbeans.modules.form.compat2.layouts.DesignAbsoluteLayout$AbsoluteConstraintsDescription">
      <AbsoluteConstraints x="375" y="172" width="50" height="-1"/>
    </Constraint>
  </Constraints>
</Component>
<Component class="javax.swing.JLabel" name="jLabel7">
  <Properties>
    <Property name="text" type="java.lang.String" value="h"/>
  </Properties>
  <Constraints>
    <Constraint
value="org.netbeans.modules.form.compat2.layouts.DesignAbsoluteLayout$AbsoluteConstraintsDescription">
      <AbsoluteConstraints x="430" y="175" width="-1" height="-1"/>
    </Constraint>
  </Constraints>
</Component>
<Component class="javax.swing.JLabel" name="jLabel8">
  <Properties>
    <Property name="background" type="java.awt.Color" editor="org.netbeans.beaninfo.editors.ColorEditor">
      <Color blue="0" green="0" red="ff" type="rgb"/>
    </Property>
    <Property name="opaque" type="boolean" value="true"/>
  </Properties>
  <Constraints>
    <Constraint
value="org.netbeans.modules.form.compat2.layouts.DesignAbsoluteLayout$AbsoluteConstraintsDescription">
      <AbsoluteConstraints x="442" y="173" width="10" height="15"/>
    </Constraint>
  </Constraints>
</Component>
<Component class="javax.swing.JToggleButton" name="Automatic">
  <Properties>
    <Property name="text" type="java.lang.String" value="Automàtic"/>
  </Properties>
  <Events>
    <EventHandler event="actionPerformed" listener="java.awt.event.ActionListener" parameters="java.awt.event.ActionEvent"
handler="AutomaticActionPerformed"/>
  </Events>
  <Constraints>
    <Constraint
value="org.netbeans.modules.form.compat2.layouts.DesignAbsoluteLayout$AbsoluteConstraintsDescription">
      <AbsoluteConstraints x="260" y="170" width="110" height="-1"/>
    </Constraint>
  </Constraints>
</Component>
<Component class="javax.swing.JLabel" name="jLabel9">
  <Properties>
    <Property name="icon" type="javax.swing.Icon" editor="org.netbeans.modules.form.editors2.IconEditor">
      <Image iconType="3" name="/copiabd/jeroni.png"/>
    </Property>
  </Properties>
  <Constraints>
    <Constraint
value="org.netbeans.modules.form.compat2.layouts.DesignAbsoluteLayout$AbsoluteConstraintsDescription">
      <AbsoluteConstraints x="320" y="60" width="140" height="70"/>
    </Constraint>
  </Constraints>
</Component>
</SubComponents>
</Form>

```

CopiaBDIG.java

```

/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package copiabd;

import java.awt.Color;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import javax.swing.Timer;

/**
 *
 * @author Dr. Carles Torras Font
 */
public class CopiaBDIG extends javax.swing.JFrame {
    private Executa executa;
    Timer timer;
    public static int comptador;

    /**

```



```

        .addComponent(clau, javax.swing.GroupLayout.DEFAULT_SIZE, 120, Short.MAX_VALUE))
        .addComponent(disc, javax.swing.GroupLayout.PREFERRED_SIZE, 36, javax.swing.GroupLayout.PREFERRED_SIZE))
        .addContainerGap());
    };
    jPanel1Layout.setVerticalGroup(
        jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(clau, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
            javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel14))
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
        .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(basedades, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
            javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel13))
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
        .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(arxiu, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
            javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel15))
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
        .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(disc, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
            javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel16))
        .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
    );

    getContentPane().add(jPanel1, new org.netbeans.lib.awtextra.AbsoluteConstraints(40, 50, -1, 120));

    Sortir.setText("Sortir");
    Sortir.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            SortirActionPerformed(evt);
        }
    });
    getContentPane().add(Sortir, new org.netbeans.lib.awtextra.AbsoluteConstraints(20, 200, 440, -1));

    Manual.setText("Manual");
    Manual.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            ManualActionPerformed(evt);
        }
    });
    getContentPane().add(Manual, new org.netbeans.lib.awtextra.AbsoluteConstraints(20, 170, 230, -1));

    temps.setText("24");
    temps.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            tempsActionPerformed(evt);
        }
    });
    getContentPane().add(temps, new org.netbeans.lib.awtextra.AbsoluteConstraints(375, 172, 50, -1));

    jLabel7.setText("h");
    getContentPane().add(jLabel7, new org.netbeans.lib.awtextra.AbsoluteConstraints(430, 175, -1, -1));

    jLabel8.setBackground(new java.awt.Color(255, 0, 0));
    jLabel8.setOpaque(true);
    getContentPane().add(jLabel8, new org.netbeans.lib.awtextra.AbsoluteConstraints(442, 173, 10, 15));

    Automatic.setText("AutomÀtic");
    Automatic.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            AutomaticActionPerformed(evt);
        }
    });
    getContentPane().add(Automatic, new org.netbeans.lib.awtextra.AbsoluteConstraints(260, 170, 110, -1));

    jLabel9.setIcon(new javax.swing.ImageIcon(getClass().getResource("/copiabd/jeroni.png"))); // NOI18N
    getContentPane().add(jLabel9, new org.netbeans.lib.awtextra.AbsoluteConstraints(320, 60, 140, 70));

    pack();
} // </editor-fold> // GEN-END: initComponents

private void basedadesActionPerformed(java.awt.event.ActionEvent evt) { // GEN-FIRST:event_basedadesActionPerformed
    // TODO add your handling code here:
} // GEN-LAST:event_basedadesActionPerformed

private void arxiuActionPerformed(java.awt.event.ActionEvent evt) { // GEN-FIRST:event_arxiuActionPerformed
    // TODO add your handling code here:
} // GEN-LAST:event_arxiuActionPerformed

private void discActionPerformed(java.awt.event.ActionEvent evt) { // GEN-FIRST:event_discActionPerformed
    // TODO add your handling code here:
} // GEN-LAST:event_discActionPerformed

private void SortirActionPerformed(java.awt.event.ActionEvent evt) { // GEN-FIRST:event_SortirActionPerformed
    System.exit(0);
} // GEN-LAST:event_SortirActionPerformed

private void ManualActionPerformed(java.awt.event.ActionEvent evt) { // GEN-FIRST:event_ManualActionPerformed
    String laclau = clau.getText();
    String labd = basedades.getText();
    String larxiu = arxiu.getText();
    String eldisc = disc.getText();

```

```

        this.executa = new Executa(laclau, labd, larxiu, eldisc, comptador);
        this.executa.execute();
    } //GEN-LAST:event_ManualActionPerformed

    private void AutomaticActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_AutomaticActionPerformed
        if(Automatic.isSelected()){
            comptador = 0;
            double lotemps = Double.parseDouble(temps.getText()); //hores
            lotemps = lotemps * 60. * 60. * 1000.; //milisegons
            int delay = (int) lotemps;
            //System.out.println(delay);
            jLabel8.setBackground(Color.green);
            ActionListener taskPerformer = new ActionListener() {
                public void actionPerformed(ActionEvent evt) {
                    comptador = comptador + 1;
                    Manual.doClick();
                }
            };
            timer = new javax.swing.Timer(delay, taskPerformer);
            timer.start();
        } else {
            jLabel8.setBackground(Color.red);
            timer.stop();
        }
    } //GEN-LAST:event_AutomaticActionPerformed

    private void clauActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_clauActionPerformed
        // TODO add your handling code here:
    } //GEN-LAST:event_clauActionPerformed

    private void tempsActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_tempsActionPerformed
        // TODO add your handling code here:
    } //GEN-LAST:event_tempsActionPerformed

    // Variables declaration - do not modify//GEN-BEGIN:variables
    private javax.swing.JToggleButton Automatic;
    private javax.swing.JButton Manual;
    private javax.swing.JButton Sortir;
    private javax.swing.JTextField arxiu;
    private javax.swing.JTextField basedades;
    private javax.swing.JPasswordField clau;
    private javax.swing.JTextField disc;
    private javax.swing.JLabel jLabel1;
    private javax.swing.JLabel jLabel2;
    private javax.swing.JLabel jLabel3;
    private javax.swing.JLabel jLabel4;
    private javax.swing.JLabel jLabel5;
    private javax.swing.JLabel jLabel6;
    private javax.swing.JLabel jLabel7;
    private javax.swing.JLabel jLabel8;
    private javax.swing.JLabel jLabel9;
    private javax.swing.JPanel jPanel1;
    private javax.swing.JTextField temps;
    // End of variables declaration//GEN-END:variables
}

```

Executa.java

```

/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

package copiabd;

import java.io.File;
import java.io.FileInputStream;
import java.io.FileNotFoundException;
import java.io.FileOutputStream;
import java.io.IOException;
import java.io.InputStream;
import java.io.PrintStream;
import java.text.DateFormat;
import java.text.SimpleDateFormat;
import java.util.Date;
import java.util.logging.Level;
import java.util.logging.Logger;
import java.util.zip.ZipEntry;
import java.util.zip.ZipOutputStream;
import javax.swing.SwingWorker;

/**
 *
 * @author Dr. Carles Torras Font
 */
public class Executa extends SwingWorker<String, String> {
    String laclau;
    String labd;
    String larxiu;
    String eldisc;
    int comptador;

    public Executa(String laclau, String labd, String larxiu, String eldisc, int comptador) {
        this.laclau = laclau;
        this.labd = labd;
    }
}

```

```

        this.larxiu = larxiu;
        this.eldisc = eldisc;
        this.comptador = comptador;
    }

    @Override
    protected String doInBackground() throws FileNotFoundException, IOException {
        // En aquesta part executo el mysqldump per recuperar les dades
        // de la base de dades.

        // ParÃ metres reals
        File test=new File(eldisc + "\\\" + larxiu + ".sql");
        // ParÃ metres de prova
        //test=new File("c:\\\\iBTWU\\" + larxiu + ".sql");

        // CÃ?pia de la base de dades ----->>
        String dumpCommand = "c:\\\\program files\\\\mysql\\\\mysql server 5.6"
            + "\\bin\\\\mysqldump " + labd + " -u " + "root" + " -p" + laclau;
        Runtime rt = Runtime.getRuntime();
        PrintStream ps = null;

        try {
            Process child = rt.exec(dumpCommand);
            ps=new PrintStream(test);
            InputStream in = child.getInputStream();
            int ch;
            while ((ch = in.read()) != -1) {
                ps.write(ch);
                //System.out.write(ch); //to view it by console
            }

            InputStream err = child.getErrorStream();
            while ((ch = err.read()) != -1) {
                //System.out.write(ch);
            }
        } catch (IOException ex) {
            Logger.getLogger(Executa.class.getName()).log(Level.SEVERE, null, ex);
        } finally {
            ps.flush();
            ps.close();
            ps = null;
        }
        return "Fet";
    }

    @Override
    protected void done() {
        // Un cop tinc l'arxiu sql creat, poso en un zip aquest arxiu
        // i tots els arxius que es referencien a la base de dades, desats
        // en la carpeta del glassfish

        // GrÃ cies a:
        // http://www.java2s.com/Code/Java/File-Input-
        // Output/Makingazipfileofdirectoryincludingitssubdirectoriesrecursively.htm

        // ParÃ metres reals
        String dir = "c:\\\\glassfish4\\\\glassfish\\\\domains\\\\domain1\\\\applications\\\\Jeroni\\\\larxiu";
        String zipFileName = eldisc + "\\\" + larxiu + ".zip";
        // ParÃ metres de prova
        //dir = "c:\\\\iBTWU";
        //zipFileName = "d:\\\\" + larxiu + ".zip";
        //System.out.println(comptador);

        File dirObj = new File(dir);
        ZipOutputStream out;
        try {
            out = new ZipOutputStream(new FileOutputStream(zipFileName));
            addDir(dirObj, out);
            out.close();
            if (comptador == 7) {
                copiabd.CopiaBDIG.comptador = 0;
                // Creo el timestamp
                DateFormat format = new SimpleDateFormat("yyyyMMdd_HHmss");
                Date diahora = new Date();
                String aran = format.format(diahora);
                //System.out.println(aran);
                zipFileName = eldisc + "\\\" + larxiu + "_" + aran + ".zip";
                //zipFileName = "d:\\\\" + larxiu + "_" + aran + ".zip";
                out = new ZipOutputStream(new FileOutputStream(zipFileName));
                addDir(dirObj, out);
                out.close();
            }
        } catch (FileNotFoundException ex) {
            Logger.getLogger(Executa.class.getName()).log(Level.SEVERE, null, ex);
        } catch (IOException ex) {
            Logger.getLogger(Executa.class.getName()).log(Level.SEVERE, null, ex);
        }
        //System.out.println("Creating : " + zipFileName);
    }

    static void addDir(File dirObj, ZipOutputStream out) throws IOException {
        File[] files = dirObj.listFiles();
        byte[] tmpBuf = new byte[1024];

        for (int i = 0; i < files.length; i++) {
            if (files[i].isDirectory()) {
                addDir(files[i], out);
                continue;
            }
        }
    }


```

```
    }  
    FileInputStream in = new FileInputStream(files[i].getAbsolutePath());  
    //System.out.println(" Adding: " + files[i].getAbsolutePath());  
    out.putNextEntry(new ZipEntry(files[i].getAbsolutePath()));  
    int len;  
    while ((len = in.read(tmpBuf)) > 0) {  
        out.write(tmpBuf, 0, len);  
    }  
    out.closeEntry();  
    in.close();  
} }  
}
```


Annex 5. Documentació generada

Durant l'execució del projecte s'ha generat una diversitat de documents, tal i com s'indica en la secció 1.4 d'aquest informe. A continuació, s'adjunta tota aquesta documentació.

A5.1.- I001. Política d'IT del CEE

	DOCUMENT INTERN	I001_140415_JEMOR
	Títol: Política de Tecnologies d'Informació a l'Escola d'Educació Especial Jeroni de Moragas – Móra d'Ebre (Tarragona). Autor: Carles Torras Font - Data: 14 d'abril de 2015	Revisió: 0

Política de Tecnologies d'Informació a l'Escola d'Educació Especial Jeroni de Moragas – Móra d'Ebre (Tarragona)

Introducció i justificació

L'Associació de Rehabilitació Jeroni de Moragas (AJEM), localitzat a Móra d'Ebre (Tarragona) és una organització que gestiona un Centre d'Educació Especial (CEE), un Taller Ocupacional (TO), un Centre Especial de Treball (ET) i una residència per persones amb necessitats especials. Fou fundat fa trenta anys i dona servei a les comarques de la Ribera d'Ebre, Priorat, Terra Alta, Baix Camp i Baix Ebre.

Aquest document, conté la política de Tecnologies d'Informació (IT) del CEE. L'àmbit d'aquesta política és el programari. Així, tot el programari dels ordinadors del CEE cal que segueixin aquestes pautes que identifica tot el *software* a utilitzar.

La creació d'una política d'IT respon a diverses necessitats. En primer lloc, donat que existeixen un cert nombre de màquines i usuaris, cal unificar al màxim possible el programari instal·lat en els diversos ordinadors. Això té diverses avantatges: (1) si un usuari sap fer funcionar tot el programari d'un ordinador, sap fer funcionar el de tots (facilitat de treballar en qualsevol ordinador), (2) es maximitza la compatibilitat de documents, (3) es facilita la gestió i el manteniment de les màquines. En segon lloc, cal tenir un control (inventari) del programari instal·lat en les màquines. En tercer lloc, la planificació del programari usat és necessari per optimitzar els recursos disponibles al CEE i complir amb la legislació vigent.

Després d'anàlitzar les necessitats del CEE, ha resultat que l'ús dels ordinadors per part dels professionals del CEE respon a un ús bàsic en el qual no es fa servir cap programa especial ni hi ha requeriments especials. És a dir, el programari bàsic usat és un processador de textos, reproductors multimèdia i navegadors web. Esporàdicament, es fa servir algun programa de creació de presentacions i d'edició bàsic d'imatges i so. I no es requereix cap marca en especial per tots aquests programes.


Per aquests motius i també donats els recursos econòmics disponibles per a IT, la conclusió és que el programari del CEE es pot basar en Programari Lliure, el qual és de codi obert (no és propietat de ningú), està en desenvolupament continu per una gran comunitat, és de màxima qualitat, seguretat i robustesa i és gratuït. Ofereix funcionalitat i usabilitat semblant a qualsevol altre programa propietari tradicional.

Aquest document de política d'IT del CEE és un document viu, que caldrà anar actualitzant i revisant al llarg del temps. Tant per adaptar-lo a les necessitats canviants del CEE com a la disponibilitat de nou programari i versions¹.

Definició del programari

Després d'anàlitzar les necessitats del CEE, s'ha classificat el programari requerit en tres blocs: sistema operatiu (SO), paquet ofimàtic (PO) i resta de programari (RP).

¹ El present document de política d'IT del CEE és fruit d'un projecte final de màster (Carles Torras Font, UOC, 2015). En l'execució del projecte, es va realitzar un complet anàlisi de requeriments, estudi de viabilitat i disseny del sistema, el qual va conduir a aquesta política. Per tant, per conèixer detalls de la justificació d'aquesta política, veure el projecte.

	DOCUMENT INTERN	I001_140415_JEMOR
	Títol: Política de Tecnologies d'Informació a l'Escola d'Educació Especial Jeroni de Moragas – Móra d'Ebre (Tarragona). Autor: Carles Torras Font - Data: 14 d'abril de 2015	Revisió: 0
		Pàgina: 2 de 4

Finalment, s'ha definit la creació de dos programes propis de gestió de la informació i serveis del CEE.

Sistema operatiu

La funcionalitat del SO és gairebé transparent pels professionals del CEE. És a dir, hi ha poca constatació que s'està usant un SO i de les funcions que fa aquest. Pel professional, només és un mitjà d'accés a les aplicacions. Així, els requeriments per part del professional és bàsicament la facilitat d'accés a les aplicacions. Històricament, el SO utilitzat al CEE són diverses versions de Microsoft® Windows® [1]. Per tant, per facilitar el canvi, es proposa un SO el més semblant possible al mencionat en quan a aspecte.

Linux Mint [2] és una distribució GNU/Linux derivada de la branca principal Debian i de la secundària Ubuntu [3] (Linux Mint usa el repositori d'Ubuntu). Debian és una de les principals distribucions, que gaudeix d'una llarga trajectòria i qualitat. Ubuntu (branca de Debian) és la distribució més usada en els darrers anys. I en l'actualitat, Linux Mint és una de les distribucions més populars, amb clar ascens i que porta incorporat un seguit de programari adient pel CEE. Finalment, constatar que Linux Mint és una de les distribucions que més s'assembla a Microsoft® Windows® en quan a aspecte. Actualment, la darrera versió estable de Linux Mint és la 17.1 Rebecca.

Els sistemes GNU/Linux no són ja una novetat. El programari GNU es va iniciar a mitjan dècada dels vuitanta, i el nucli Linux a començaments dels noranta. Linux es recolza en tecnologia provada de UNIX, amb més de quaranta anys d'història.

Per tots aquests motius, **Linux Mint** (versió en català) és el SO elegit com a principal pel CEE. Utilitza diverses llicències lliures, essent la majoritària la GNU/GPL v3 [4].

Donat que el CEE, actualment, disposa de diverses màquines les quals portaven incorporat alguna versió de Microsoft® Windows® (per tant, amb llicència) i que aquest SO és el de referència històricament al CEE, es defineix **Microsoft® Windows®** com a SO secundari del CEE. Així, en casos justificats, el SO d'una màquina podrà usar Microsoft Windows amb alguna de les llicències actuals.

Paquet ofimàtic


En quan a paquet ofimàtic i en entorn lliure, existeixen dos paquets principals: OpenOffice [5] i LibreOffice [6]. Els dos paquets ofereixen una gran qualitat, seguretat, maduresa, funcionalitat i robustesa. Tanmateix, en els darrers anys, OpenOffice s'ha vist involucrat en un procés de canvi que ha portat a LibreOffice a adquirir certa avantatge en quan a estat de desenvolupament i general del projecte. Per aquest motiu, s'elegeix **LibreOffice** com a paquet ofimàtic principal i únic pel CEE. Actualment, la darrera versió estable és la 4.4.2.

LibreOffice incorpora aplicacions de tractament de text, full de càlcul, creació de presentacions, etc. Conté una àmplia funcionalitat amb PDF i conté total funcionalitat en català (tant de diccionaris com de llenguatge d'interfície).

LibreOffice es distribueix sota la llicència lliure Mozilla Public License v2.0 [7].

Resta de programari

S'han identificat com a programes necessaris un editor d'imatges, un editor de so i reproductors multimèdia. Després del corresponent estudi d'opcions, s'han elegit els següents programes (tots ells PL):

	DOCUMENT INTERN	I001_140415_JEMOR
	Títol: Política de Tecnologies d'Informació a l'Escola d'Educació Especial Jeroni de Moragas – Móra d'Ebre (Tarragona). Autor: Carles Torras Font - Data: 14 d'abril de 2015	Revisió: 0
		Pàgina: 3 de 4

- Paquet de tractament d'imatges: Gimp [8]. Cobreix les necessitats bàsiques que tenen els usuaris, tot i que el programa ofereix una funcionalitat molt extensa. Programa de maduresa alta. Disponible per Windows® i Mint. Versió actual: 2.8.14.
- Paquet de tractament de so: Audacity [9]. Cobreix les necessitats bàsiques que tenen els usuaris. Programa de maduresa alta. Disponible per Windows® i Mint. Versió actual: 2.1.0
- Reproductor i còdecs multimèdia. Un reproductor i còdecs ja estan inclosos en el propi sistema operatiu Linux Mint. Tanmateix, poden ser insuficients. Per tant, es complementarà amb: MPlayer i SMPlayer, VLC, FFMpeg i GStreamer i ubuntu-restricted-extras. Cal considerar que no tots els còdecs corresponen a PL (però sí gratuït).


Programari propi

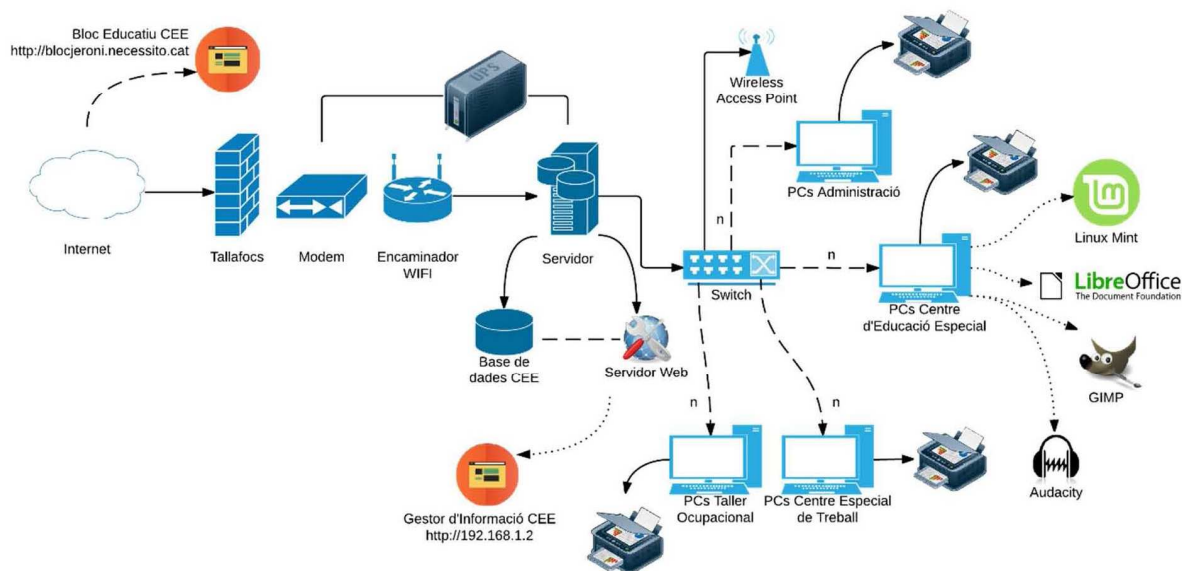
Actualment, al CEE només es treballa amb fitxers. Tota la informació està desada en els mateixos (típicament, arxius de processador de textos). No existeix cap sistema de gestió de dades (des del punt de vista electrònic). Així, es proposa desenvolupar i implementar un **sistema de gestió de la informació**, tot i aprofitant els recursos d'IT. Aquest ha de permetre millorar molt l'eficiència en gestió.

Així, és posarà en marxa una base de dades i un gestor de base de dades. Al mateix temps, es crearà una interfície web de connexió per l'usuari que li permeti la gestió de les mateixes. S'hi ha de gestionar:

- Llistat d'alumnes amb les següents dades: nom i cognoms, NIF, data naixement, adreça, població, telèfons, noms pare i mare, correu electrònic, any entrada al centre, educació compartida o no, tutora.
- Diagnòstics de les discapacitats dels alumnes: diagnòstic, professional o centre on s'ha fet i data.
- Medicacions dels alumnes: medicació, qui l'ha receptat i qui fa el seguiment.
- Professionals del CEE amb les següents dades: nom i cognoms, NIF, categoria i horari.
- Aules del CEE, nom, configuracions i els seus horaris.
- Registre d'assistència dels alumnes.
- Informes dels alumnes.
- Registre d'incidències.
- Reunions: data, hora, assistents, lloc, temes tractats, acords, propera reunió.
- Normes de convivència i funcionament del CEE.
- Fotografies. Adequació de la mida. Arxiu en base a l'aula i etiquetes.

Adicionalment, es crearà un **Bloc Educatiu**. Es pretén crear un bloc però no amb una finalitat externa (pel públic extern) sinó com una eina de treball pels alumnes del CEE. Es pretén que els alumnes puguin crear continguts i eventualment, mostrar-los als seus pares, amics, etc.

	DOCUMENT INTERN	I001_140415_JEMOR
	Títol: Política de Tecnologies d'Informació a l'Escola d'Educació Especial Jeroni de Moragas – Móra d'Ebre (Tarragona). Autor: Carles Torras Font - Data: 14 d'abril de 2015	Revisió: 0
		Pàgina: 4 de 4



Infraestructura d'IT projectada d'AJEM. Diagrama de components.

Referències

1. Microsoft. *Pàgina web oficial de Microsoft Windows*. 2015; Available from: <http://windows.microsoft.com/es-es/windows/home>.
2. Mint, L. *Distribució Linux Mint*. 2015; Available from: <http://www.linuxmint.com/>.
3. Canonical. *Distribució Ubuntu*. 2015; Available from: <http://www.ubuntu.com/>.
4. GNU, *GNU General Public License v3*. 2007. <http://www.gnu.org/copyleft/gpl.html>
5. OpenOffice. *Projecte OpenOffice*. 2015; Available from: <http://www.openoffice.org/ca/>.
6. LibreOffice. *Projecte LibreOffice*. 2015; Available from: <https://ca.libreoffice.org/>.
7. Mozilla, *Mozilla Public License 2.0*. 2012. <https://www.mozilla.org/MPL/>
8. GIMP, C.d.d.d. *GIMP. The GNU Image Manipulation Program*. 2015; Available from: <http://www.gimp.org/>.
9. d'Audacity, C.d.d. *Projecte Audacity*. 2015; Available from: <http://audacity.sourceforge.net/>.

A5.2.- I002. Guia d'ús del Bloc Educatiu

	DOCUMENT INTERN	I002_180415_JEMOR
	Títol: Guia d'ús del Bloc Educatiu "Jeroni de Moragas" Autor: Carles Torras Font Data: 18 d'abril de 2015	Revisió: 2
		Pàgina: 1 de 4

Guia d'ús del Bloc Educatiu "Jeroni de Moragas"

- S'accedeix al bloc mitjançant un navegador web (Firefox v. 36 o superior, Chrome v. 41 o superior, Internet Explorer v. 11 o superior). Direcció: <http://blocjeroni.necessito.cat>
- Quan s'entra, es visualitzen automàticament els darrers missatges:



- Els missatges es poden **filtrar** per aula, data o ambdues coses. Per fer-ho, clicar a l'opció "Filtra missatges", ubicada al menú inferior.
 - A l'accedir a la pàgina de filtre, cal seleccionar quin tipus de filtre es vol aplicar (clicant a un dels tres *radiobuttons*). Si es filtra per aula, cal seleccionar-ne una a través del desplegable. Si es filtre per calendari, cal indicar la data inicial i la final seleccionant una data des del calendari (que s'obre automàticament al clicar a cadascun dels dos camps de text). Finalment, prémer "Executar canvis" per filtrar o "tornar" per cancel·lar i tornar a la pàgina d'inici.

Filtra missatges

Filtrat per calendari
 Filtrar per aula
 Filtrar per calendari i aula

Elegeix una aula

Gimnàs

Selector de data. Des de: Fins:

Executar canvis

Tornar

April 2015

Su	Mo	Tu	We	Th	Fr	Sa
			1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30		

	DOCUMENT INTERN	I002_180415_JEMOR
	Títol: Guia d'ús del Bloc Educatiu "Jeroni de Moragas" Autor: Carles Torras Font Data: 18 d'abril de 2015	Revisió: 2
		Pàgina: 2 de 4

- Per **inserir** un missatge (des de la pàgina principal), cal clicar sobre el text "Introdueix un missatge". Al fer-ho, s'obre un desplegable que permet introduir les dades del missatge. Aquestes són:
 - El cos del missatge (obligatori).
 - L'usuari (obligatori). Només els usuaris autoritzats (donats d'alta al sistema) poden introduir missatges.
 - La data i hora (optatiu). Quan es clica damunt el quadre de text, es mostra un calendari i selectors per especificar data i hora. Si no s'especifica, el sistema desarà la imatge amb la data de càrrega de la fotografia al sistema.
 - Fotografies (opcional). Clicant al botó "Trieu els fitxers" es poden seleccionar una o més fotografies. Cal tenir en compte que si es seleccionen diverses fotografies i tenen una mida elevada, el procés de càrrega pot trigar estona (en funció també de la connexió a Internet).
 - Clicant a un dels dos *radiobutton* es pot especificar si, posteriorment, es voldrà especificar títols a les fotografies.

Finalment, clicar a "envia'l" per enviar el missatge.




Si no s'ha especificat afegir títol a les fotografies, el procés haurà acabat i es mostraran els missatges del bloc. Si s'ha especificat afegir-los, es demanarà especificar els títols. Com mostra la següent figura, caldrà posar els noms de cada fotografia i clicar a "Afegir títols". Per cancel·lar (s'introduiran les fotografies sense títols), clicar a "tornar".

Introdueix els títols per les fotos que has introduït:



Títol fotografia 1:

- Els missatges també es poden **eliminar**. Cada usuari pot eliminar només els seus missatges i d'un en un. L'administrador pot eliminar els seus i els de tots. Per eliminar un missatge:

	DOCUMENT INTERN	I002_180415_JEMOR
	Títol: Guia d'ús del Bloc Educatiu "Jeroni de Moragas" Autor: Carles Torras Font Data: 18 d'abril de 2015	Revisió: 2
		Pàgina: 3 de 4

- o Des de la pàgina principal, clicar damunt el nº de missatge a eliminar (darrer número que figura a la barra verda de cada missatge). Es mostrarà la pantalla d'eliminació.
- o En la pantalla d'eliminació, introduir el nom d'usuari i clicar "Elimina missatge" (el nº de missatge ja s'introduirà automàticament a partir de la selecció anterior). Per cancel·lar, prémer "Tornar".

Eliminar missatges

Usuari amb permisos per gestionar

Nom d'usuari

Missatge a eliminar: 144

Elimina missatge

- Els missatges també es poden **modificar**. Cada usuari pot modificar només els seus missatges i d'un en un. L'administrador pot modificar els seus i els de tots. Per modificar un missatge:
 - o Des de la pàgina principal, clicar damunt el nº de missatge a modificar (darrer número que figura a la barra verda de cada missatge). Es mostrarà la pantalla de modificació.
 - o En la pantalla de modificació, introduir el nom d'usuari, modificar els camps pertinents (text i/o data) i clicar "Modificar missatge" (el nº de missatge ja s'introduirà automàticament a partir de la selecció anterior). Per cancel·lar, prémer "Tornar".

Modificar missatges

Usuari amb permisos per modificar

Nom d'usuari

Missatge a eliminar: 124

Text

L'embat

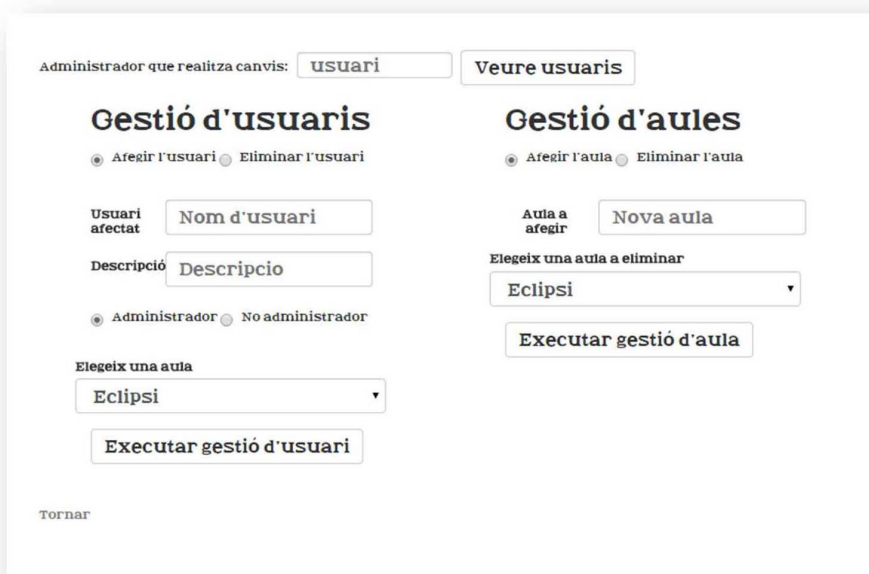
Data: 2015-04-22 13:23:16.0

Modificar missatge

- El Bloc permet **gestionar** els **usuaris** i les **aules**. Per fer-ho, des de la pàgina principal, clicar a "Gestió d'usuaris/aules". A la pàgina de gestió, es poden realitzar tres accions: visualitzar usuaris, gestionar usuaris o gestionar aules. Per qualsevol acció, cal tenir drets d'administració. I per tant, per qualsevol acció cal omplir el camp "Administrador que realitza canvis".
 - o Gestió d'aules. Es poden afegir o eliminar aules (selecció mitjançant el *radiobutton*). Si s'afegeix, cal indicar el nom al quadre de text. Si s'elimina, cal seleccionar l'aula del desplegable. No es podran eliminar aules si existeixen missatges a nom d'aquella aula (no es permeten missatges orfes, primer, caldria eliminar els missatges). Abans de clicar al botó "Executar gestió d'aula" per procedir, cal indicar el nom d'usuari a "Administrador que realitza canvis".

	DOCUMENT INTERN	I002_180415_JEMOR
	Títol: Guia d'ús del Bloc Educatiu "Jeroni de Moragas" Autor: Carles Torras Font Data: 18 d'abril de 2015	Revisió: 2
		Pàgina: 4 de 4

- o Gestió d'usuaris. Es poden afegir o eliminar usuaris. Per eliminar-lo, només cal completar la informació als quadres de text "Administrador que realitza canvis" i "Usuari afectat" (usuari a eliminar). Finalment, cal clicar a "Executar gestió d'usuari". Per afegir-lo, a banda de l'anterior, cal especificar una descripció de l'usuari (quadre "Descripció"), cal indicar si tindrà drets o no administrador (*radiobutton*) i a quina aula pertany (seleccionar-la del desplegable). El nom d'usuari s'encriptarà i no es podrà visualitzar el nom original. Per aquest motiu, per reconèixer qui és l'usuari, cal fer servir el camp de descripció.



En qualsevol acció que es realitzi (afegir usuaris, afegir missatges, eliminar missatges, etc.), l'aplicació mostra un missatge indicant si s'ha realitzat correctament o no.



A5.3.- I003. Guia d'ús del Gestor d'Informació

	DOCUMENT INTERN	I003_200415_JEMOR
	Títol: Guia d'ús del Gestor d'Informació del CEE Jeroni de Moragas Autor: Carles Torras Font - Data: 20 d'abril de 2015	Revisió: 1
		Pàgina: 1 de 10

Guia d'ús del Gestor d'Informació del CEE Jeroni de Moragas

Definicions generals	1
Definicions d'objectes a gestionar	3
Guia d'ús	4
Gestió d'un ítem genèric	4
Gestió de les classes	6
Gestor fotogràfic	8
Dades creades fictícies pel test del gestor	10

Definicions generals

- El gestor està dividit en **tres seccions**: (1) gestió del director, (2) gestió del tutor i (3) gestor fotogràfic.
- Les **accions** a realitzar dins del gestor **depenen del rol** i si un professional és **tutor o no**. Hi ha els següent rols: administrador/a, director/a, mestre/a, fisioterapeuta, logopeda, educador/a.



- El **gestor del director** permet:
 - Gestionar **professionals**. Afegir, eliminar, modificar i llistar. Es pot modificar tot menys la clau d'accés (només el propi usuari pot modificar la clau: és personal i per responsabilitat pròpia de les accions).
 - Gestionar **alumnes**. Afegir, eliminar, modificar i llistar.
 - Gestionar **aules i classes**. Es defineix **aula** com aquell espai físic que roman any rere any i que acull les **classes** que cada any són diferents. Per exemple, "Eclipsi" és el nom d'una aula (espai físic que s'hi accedeix per unes portes, etc.) i que cada curs acadèmic acull una classe diferent. Per exemple, el curs 2012/13 va acollir una classe formada per un tutor, uns

	DOCUMENT INTERN	I003_200415_JEMOR
	Títol: Guia d'ús del Gestor d'Informació del CEE Jeroni de Moragas Autor: Carles Torras Font - Data: 20 d'abril de 2015	Revisió: 1
		Pàgina: 2 de 10

alumnes, etc., que va existir aquell curs acadèmic i que mai tornarà a existir. La classe té un identificador únic (no es tornarà a repetir més) anomenat: Eclipsi, curs 2012/13. El gestor permet crear, eliminar, modificar i llistar **classes**. Un cop la classe està creada, es pot gestionar els tutors i alumnes que en formen part. Aquests paràmetre també es poden modificar en qualsevol moment. Les **aules** es creen a partir del nom de l'aula i el curs. Els noms d'aules es poden crear i eliminar.

- Exemple: es pot crear l'aula "Eclipsi, 2015-16" a partir del nom "Eclipsi" i el curs "2015-16".
 - Gestionar normes. Afegir, eliminar, modificar i llistar.
- El **gestor del tutor** permet:
 - Gestionar faltes d'assistència. Afegir, eliminar i llistar.
 - Gestionar incidències. Afegir, eliminar i llistar.
 - Gestionar informes. Afegir, eliminar, modificar i llistar.
 - Gestionar reunions. Afegir, eliminar, modificar i llistar.
- El **gestor fotogràfic**:
 - Gestionar fotografies. Afegir, modificar, eliminar.
 - Visualitzar el catàleg de fotografies i possibilitat de filtratge per aula, data i/o etiqueta.
- En **funció del rol** de l'usuari, es pot accedir a una o més seccions. El rol de director permet accedir a totes les seccions, el rol de tutor permet accedir a la secció 2 i 3, i el rol d'usuari normal només permet accedir al gestor fotogràfic.
- Un **usuari tutor** només podrà gestionar faltes d'assistència, incidències, informes i reunions dels alumnes que estiguin donats d'alta en classes seves.
- Un **usuari director** podrà llistar totes les assistències, incidències, informes i reunions, però no podrà gestionar. Només podrà fer-ho l'usuari tutor qui les va crear.
- Totes les **accions** realitzades per qualsevol usuari són **enregistrades**: usuari, acció realitzada i quan (data i hora).
- L'aplicació porta incorporada un primer sistema de **comunicació amb l'usuari**. Aquest consisteix en informar-lo després d'una acció de creació, modificació o eliminació. S'informa si s'ha efectuat amb èxit o no. En una segona fase, es podrà millorar aquest sistema incloent una descripció del motiu pel qual no s'ha realitzat amb èxit l'operació, si és que ha estat així.
- Principis i bases de disseny a tenir en compte:
 - **Principi de l'identificador**. Dels diversos ítems que permeten modificacions (per exemple, un professional), es pot modificar tot menys l'identificatiu (per exemple, el NIF en el cas del professional). En aquest manual, per cada ítem s'identifica quin és l'identificatiu. Un identificatiu no es pot modificar perquè es considera que es tracta d'un nou ítem (es pot crear i eliminar).
 - **Principi de l'orfanat**. És possible que en diversos casos que es vulgui realitzar una eliminació no es pugui completar. Per exemple, no es pot eliminar un professional si aquest té classes associades (si s'eliminés, la classe romandria orfe de tutor). O per exemple, amb molta probabilitat, un alumne mai podrà ser eliminat ja que roman un registre de les vegades que s'ha visualitzat informació sobre el mateix.

	DOCUMENT INTERN	I003_200415_JEMOR
	Títol: Guia d'ús del Gestor d'Informació del CEE Jeroni de Moragas Autor: Carles Torras Font - Data: 20 d'abril de 2015	Revisió: 1
		Pàgina: 3 de 10

Definicions d'objectes a gestionar

En la següent llista s'identifiquen els diversos objectes que es poden gestionar juntament amb els seus atributs. Es subratllen els identificadors de cada objecte.


- *Professional.* NIF, nom, cognoms, tutor (si ho és o no), horari del professional (arxiu adjunt), clau d'accés, usuari actiu (si ho és o no)¹ i nom de categoria.
- *Alumne.* NIF, nom, cognoms, data de naixement, nom del pare, nom de la mare, nom de la parella del pare, nom de la parella de la mare, nom del tutor, adreça, adreça electrònica, 4 camps de telèfon, número de la targeta sanitària, compartit (si l'alumne es comparteix amb un altre centre o no), any d'entrada al CEE, 4 camps de diagnòstic mèdic, 4 camps de medicació, programa (arxiu adjunt).
- *Classes.* Les classes es defineixen inequívocament per la suma d'un nom aula i d'un curs acadèmic (veure l'apartat de definicions generals, per exemple: Logopèdia, curs 2015/16). Ambdós ítems sempre es gestionen a partir de llistes que es poden modificar (per exemple, existeix una llista pre-definida de noms d'aula que es modificar). Cada classe conté: nom curs, llista de professionals tutors assignats, llista d'alumnes que l'integren i horari de la classe (arxiu adjunt).
- *Faltes d'assistència.* Alumne que ha faltat, data, motiu, professional tutor i número d'identificació de la falta.
- *Reunions.* Alumne sobre el qual s'ha fet la reunió, professionals del CEE que han assistit, altres assistents, data i hora, número de la reunió predecessora, acta o contingut de la reunió (arxiu adjunt) i número d'identificació de la reunió.
- *Informes.* Alumne del qual es realitza l'informe, professionals involucrats, data, títol, informe (arxiu adjunt) i número d'identificació de l'informe.
- *Incidències.* Nom de la incidència, data i hora, professional del CEE que la reporta, alumnes involucrats, informe de la incidència (arxiu adjunt) i número d'identificació de la incidència.
- *Normes.* Títol, text de la norma (arxiu adjunt) i número d'identificació de la norma.
- *Fotografia.* 4 etiquetes que defineixen la fotografia (substitueixen el títol i els camps són opcionals), aula a la que pertany, llista d'alumnes associats, data, fotografia (arxiu adjunt) i número d'identificació de la fotografia.

Per tal de donar compliment a la Llei de Protecció de Dades, a banda dels anteriors ítems (propis de gestió), el sistema també emmagatzema els accessos a la base de dades, les accions realitzades, per qui i quan:

- *Accés.* Professional que accedeix al gestor, data i hora, acció que realitza i número d'identificació de la mateixa.

Llistat d'accions disponible en l'accés al gestor d'informació del CEE				
acces	afegir_professional	modificar_professional	eliminar_professional	l·listar_professional
afegir_alumne	modificar_alumne	eliminar_alumne	l·listar_alumne	afegir_nomaula
eliminar_nomaula	assignar_alumnes_aula	afegir_aula	modificar_aula	eliminar_aula
l·listar_aula	afegir_norma	modificar_norma	eliminar_norma	l·listar_norma
afegir_assist	eliminar_assist	l·listar_assist	afegir_incidencia	eliminar_incidencia
l·listar_incidencia	afegir_informes	modificar_informes	eliminar_informes	l·listar_informes

¹ Si un professional esdevé baixa del CEE i si ha realitzat activitat al Gestor, no es podrà eliminar del sistema ja que l'històric d'activitat i dades ha de romandre. En conseqüència, per impedir-li l'accés i evitar que realitzi cap més acció, l'usuari amb rol de director o administrador pot marcar-lo com a usuari "no actiu".

	DOCUMENT INTERN	I003_200415_JEMOR
	Títol: Guia d'ús del Gestor d'Informació del CEE Jeroni de Moragas Autor: Carles Torras Font - Data: 20 d'abril de 2015	Revisió: 1
		Pàgina: 4 de 10

afegir_reunions	modificar_reunions	eliminar_reunions	l·listar_reunions	afegir_foto
modificar_foto	eliminar_foto	l·listar_foto		

Guia d'ús

- **S'accedeix** al gestor mitjançant un navegador web (Firefox v. 36 o superior, Chrome v. 41 o superior, Internet Explorer v. 11 o superior). Direcció: <http://192.168.2.2/Gestor> La versió de proves està accessible a la direcció: <http://jeroni.necessito.cat>
- **Per entrar-hi**, cal validar-se amb un nom d'usuari i contrasenya. El nom d'usuari és el NIF i el nom de contrasenya és personalitzable (pel primer accés, l'administrador del sistema facilità una clau).
- Un cop dins del sistema, es visualitzen **tres apartats**: capçalera, opcions d'usuari i ítems de gestió:



- En les **opcions d'usuari** s'indica el nom i NIF del mateix, que s'ha iniciat sessió i la fotografia. Les opcions són canviar la fotografia, canviar la clau d'accés i tancar la sessió. Cada usuari és responsable de les accions que dugui a terme, les quals són registrades. Per tant, és important que mantingui la confidencialitat de la clau d'accés (personal i intransferible) i que tanqui la sessió un cop finalitzada la feina (i addicionalment, es recomana també tancar el navegador).
- En la zona **d'ítems de gestió**, poden haver-hi fins a tres seccions que estaran disponibles o no en funció del rol de l'usuari: gestor del director, gestor del tutor i gestor fotogràfic (veure les definicions generals). Sota cada títol de cada secció es l·listen els ítems de gestió disponible. El funcionament de tots ells és molt semblant.

Gestió d'un ítem genèric

Tots els ítems es gestionen d'una manera molt similar. Com a exemple, es mostra a continuació la gestió de reunions (secció gestor del tutor). Per gestionar-les, cal fer clic sobre el títol mateix, i llavors, es despleguen les opcions de gestió:



Existeixen 2 opcions: l·listar, modificar o eliminar i afegir. Si es selecciona la primera opció i es clica al botó "seleccionar", s'accedeix a la següent pantalla:

	DOCUMENT INTERN	I003_200415_JEMOR
	Títol: Guia d'ús del Gestor d'Informació del CEE Jeroni de Moragas Autor: Carles Torras Font - Data: 20 d'abril de 2015	Revisió: 1
		Pàgina: 5 de 10

Llistat de reunions

	id	Alumne	Assistents	Data i hora	Informe	Anterior	Professionals	
Modificar	Eliminar	1	Alba Ferm, 23565466J	Sons pares	2015-05-04 03:15:00.0	Reunio.docx	0	Queralt Isona, 50000000A;
Modificar	Eliminar	2	Miki Puig, 45465564G	Ningú	2015-05-07 04:00:00.0	Reunio.docx	0	Joan Carrasco, 20000000A;

Tornar

Aquesta pantalla llista les reunions introduïdes. En la llista es mostren totes les reunions d'aquell professional o si l'usuari té el rol de director o administrador totes les existents. Per cada reunió es mostra tota la informació. També l'arxiu d'informe de la reunió al qual, si s'hi fa clic a sobre, s'obre o es descarrega. Fent clic al botó "tornar" es retorna a la pàgina principal.

Al principi de cada línia (registre) existeixen dos botons, un per eliminar l'entrada i l'altre per modificar-la. Si es fa clic al botó "eliminar", aquesta s'esborrarà si és possible (segons el criteri de l'orfanat explicat a sobre). Si es fa clic a "modificar", s'accedirà a una altre pantalla per fer-ho.

Modificar reunions

Alumne	Professional
Alba Ferm Murts, NIF: 23565466J Aria Marc Pem, NIF: 77676765G	Joana Vilatorrada Mer, NIF: 00000000A Aurembiaix Martínez Olesa, NIF: 10000000A Joan Carrasco Fox, NIF: 20000000A Manel Formosa Vilella, NIF: 30000000A
Assistents: <input type="text" value="Ningú"/>	
Data i hora: <input type="text" value="2015-05-07 06:00:00"/>	
Informe de la reunió: <input type="text" value="Reunió3.docx"/>	
Nou informe de la reunió: <input type="button" value="Navega..."/> No s'ha seleccionat cap fitxer.	
Reunió anterior: <input checked="" type="radio"/> No n'hi ha cap. <input type="radio"/> N'hi ha: <input type="text" value="2"/>	
<input type="button" value="Modifica"/>	

Aquesta pantalla inclou tots els elements que es poden configurar de l'objecte "reunió":

- **Alumne.** Es mostra la llista d'alumnes que el professional té assignats i es mostra l'alumne seleccionat per aquesta reunió. Es pot canviar seleccionant un altre alumne (només un).
- **Professional.** Es mostra la llista de professionals del CEE i el/s seleccionat/s per aquesta reunió. Es pot canviar seleccionant altres professional (un o diversos). Cal tenir en compte que sempre quedarà seleccionat automàticament l'usuari tutor. Exemple: si l'usuari és en Joan Carrasco, i aquest selecciona només la Joana Vilatorrada, el sistema emmagatzemarà per aquesta reunió el Joan i la Joana. O si no selecciona ningú, el sistema seleccionarà automàticament el Joan.
- **Assistents.** A part de l'alumne i els professional, poden haver-hi altres assistents a la reunió (per exemple, els pares). Aquests s'introdueixen en el camp de text lliure (opcional).
- **Data i hora.** Quan es clica damunt el camp, es mostra un calendari que permet fer la selecció de la data i l'hora.

	DOCUMENT INTERN	I003_200415_JEMOR
	Títol: Guia d'ús del Gestor d'Informació del CEE Jeroni de Moragas Autor: Carles Torras Font - Data: 20 d'abril de 2015	Revisió: 1
		Pàgina: 6 de 10

Data i hora 2015-05-07 06:00:00

Informe de **May 2015**

Nou informe

Reunió anterior

Modifica

Su	Mo	Tu	We	Th	Fr	Sa
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
31						

Time 12:00:00

Hour

Minute

Second

Now Done

- **Informe de la reunió.** Es mostra l'arxiu actual de la reunió, el qual es pot clicar per obrir-lo. Si no es vol canviar, no cal fer res. Si es vol canviar, es selecciona el nou arxiu des del botó "navega". Finalment, es pot modificar si aquesta reunió en té una d'antecedent o no. Si es clica que sí, cal seleccionar quina.

Finalment, per afegir una reunió, es clica sobre el botó tipus radio "afegir". Amb això, es desplega un nou formulari amb els camps comentats anteriorment (apartat de modificar) els quals tenen els mateixos condicionants i que calen omplir. Finalment, cal fer clic al botó "executar canvis":

Gestor de reunions

Seleccionar Llistar, modificar o eliminar Afegir

Alumne Professional

Joan Albi Pulla, NIF: 46571614M Joana Vilatorrada Mer, NIF: 00000000A
 Joan Pere Benet, NIF: 69767809E Aurembiaix Martínez Olesa, NIF: 10000000A
 Juli Cesc Paquí, NIF: 65156425D Joan Carrasco de Foix, NIF: 20000000A
 Miki Puig Ferré, NIF: 45465564G Manel Formosa Vilella, NIF: 30000000A

Assistents

assistents

Data/hora

Informe de la reunió: No s'ha triat cap fitxer

Reunió anterior: No n'hi ha cap. N'hi ha:

Gestió de les classes

Així, tots els apartats de gestió funció de la mateixa manera, a excepció del de **gestió de les classes**, que es comenta a continuació. Un cop es clica damunt "gestor de classes", es mostra el desplegable d'opcions com en la resta de casos. Però quan es selecciona l'opció de llistar, modificar o eliminar, existeix una altra opció que és "assignar". Aquesta permet assignar tutors i alumnes a la classe. Això no es realitza quan es crea, sinó en aquest apartat.

	DOCUMENT INTERN	I003_200415_JEMOR
	Títol: Guia d'ús del Gestor d'Informació del CEE Jeroni de Moragas Autor: Carles Torras Font - Data: 20 d'abril de 2015	Revisió: 1
		Pàgina: 7 de 10

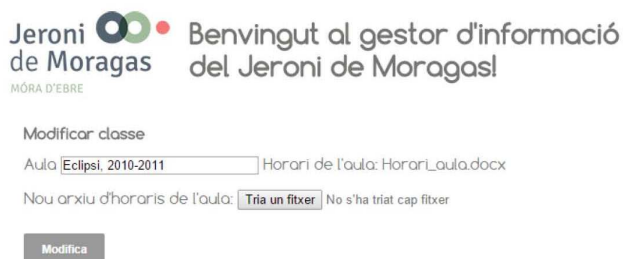
Listat d'aules

Clicar damunt el nom de la classe per veure els seus tutors i alumnes

	Nom	Horari			
<table border="1"> <tr><td>Modificar</td><td>Assignar</td><td>Eliminar</td></tr> </table>	Modificar	Assignar	Eliminar	Eclipsi, 2010-2011	Horari_aula.docx
Modificar	Assignar	Eliminar			
<table border="1"> <tr><td>Modificar</td><td>Assignar</td><td>Eliminar</td></tr> </table>	Modificar	Assignar	Eliminar	Eclipsi, 2011-2012	
Modificar	Assignar	Eliminar			
<table border="1"> <tr><td>Modificar</td><td>Assignar</td><td>Eliminar</td></tr> </table>	Modificar	Assignar	Eliminar	Eclipsi, 2025-2026	
Modificar	Assignar	Eliminar			
<table border="1"> <tr><td>Modificar</td><td>Assignar</td><td>Eliminar</td></tr> </table>	Modificar	Assignar	Eliminar	Logopèdia, 2025-2026	Horari_aula.docx
Modificar	Assignar	Eliminar			

Tornar

L'apartat d'afegir una classe inclou definir el conjunt nom d'aula i curs i afegir l'arxiu d'horari de la classe. Tanmateix, per assignar-hi alumnes i tutors, un cop s'ha creat l'aula, cal fer clic a "Assignar". Això es pot realitzar tant per assignar-los la primera vegada un cop s'ha creat la classe o per realitzar modificacions posterior. Així, l'opció "modificar" general d'aules només permet modificar els ítems que s'han definit en la creació de la classe:



Jeroni de Moragas MÓRA D'EBRE **Benvingut al gestor d'informació del Jeroni de Moragas!**

Modificar classe

Aula: Horari de l'aula:

Nou arxiu d'horaris de l'aula: No s'ha triat cap fitxer

Quan es fa clic a "Assignar", es mostra la següent pantalla:



Jeroni de Moragas MÓRA D'EBRE **Benvingut al gestor d'informació del Jeroni de Moragas!**

Gestió d'alumnes i tutors de la classe Eclipsi, 2010-2011

Llista d'alumnes

- Aria Marc Perri, NIF: 77676765G
- Gili Flix Pered, NIF: 83976345H
- Joan Albi Pulla, NIF: 46571614M
- Joan Pere Benet, NIF: 69767809E
- Juli Cesc Paqui, NIF: 65156425B
- Karl Marx Petra, NIF: 23466266H
- Llum Over Kilmi, NIF: 71246146W
- Miki Puig Ferre, NIF: 45465564G

Llista d'alumnes de la classe

- Alba Fern Murts, NIF: 23565466J
- Gila Munt Murts, NIF: 68554255H

Llista de tutors

- Joana Vilatorrada Mer, NIF: 00000000A

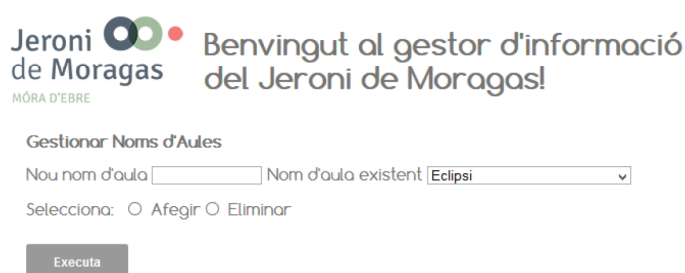
Llista de tutors de la classe

- Joan Carrasco Foix, NIF: 20000000A
- Queralt Isona Salvadó, NIF: 50000000A

Aquesta conté el nom de la classe i dos grups de llistats: la llista d'alumnes i la llista d'alumnes de la classe i la llista de tutors i la llista de tutors assignats a la classe. Per cada grup, en la primera columna es mostren tots els alumnes/tutors de la base de dades del CEE. En la segona, els alumnes/tutors assignats a la classe. Si s'accedeix al tràmit just després de crear la classe, el segon desplegable estarà buit. Mitjançant els dos botons centrals amb fletxes, es poden afegir o treure alumnes de la classe (prèvia selecció).

	DOCUMENT INTERN	I003_200415_JEMOR
	Títol: Guia d'ús del Gestor d'Informació del CEE Jeroni de Moragas Autor: Carles Torras Font - Data: 20 d'abril de 2015	
	Revisió: 1 Pàgina: 8 de 10	

La gestió d'aules permet afegir o eliminar aquests noms, els quals són posteriorment usats per definir les classes, com s'ha comentat anteriorment. Per exemple, a partir de "Logopèdia", es pot generar la classe única "Logopèdia, curs 2012/13". Cal tenir en compte que no es podran eliminar noms d'aules que tinguin classes associades. Per afegir un nom d'aula cal escriure el nom al quadre de text "Nou nom d'aula", clicar al botó-ràdio "Afegir" i prémer el botó "Executa". Per eliminar un nom d'aula, cal seleccionar un nom del desplegable "Nom d'aula existent", clicar al botó-ràdio "Eliminar" i prémer el botó "Executa".



Nota final. En funció del definit en aquest apartat, cada professional tutor que es connecti al gestor, en la secció "Gestió del tutor" tindrà definit el seu marge d'actuació. Per exemple, si a un mestre tutor no se li ha assignat cap classe en aquest apartat, no tindrà cap opció per gestionar reunions, informes, etc. Com a tutor que pugui estar definit, tindrà accés a la secció "Gestió del tutor", però sense opcions de gestió. Si un professional no és tutor, ni tindrà accés a la secció "Gestió del tutor", ni se li podran assignar classes.

Gestor fotogràfic

Finalment i en relació al **gestor fotogràfic**, esmentar tres aspectes:

- L'opció "l·listar", en aquest cas, s'anomena "catàleg".
- En l'apartat d'afegir, quan es carrega una fotografia, la mida d'aquesta és transformada automàticament a una mida d'amplada de 1020 px (i l'alçada corresponent per mantenir la relació original ampla/alt). Això es realitza per equilibrar la qualitat de la imatge i el cost d'espai per desar els arxius d'imatge.
- Quan s'accedeix al catàleg es mostren les fotos desades. Aquestes es poden filtrar fent clic a "filtre". Quan es realitza, es mostra el formulari de filtre el qual permet 4 tipus de filtre: per camp de text lliure (es cerquen coincidències en les etiquetes de les fotografies), per data, per alumne o per aula. Si es vol aplicar un filtre (se'n poden aplicar al mateix temps, un, dos, tres o quatre), cal primer seleccionar-lo i després omplir el criteri de filtre. Per exemple, en la següent figura es mostra que s'ha seleccionat el filtre de calendari i es seleccionen les dates mitjançant el calendari:

	DOCUMENT INTERN	I003_200415_JEMOR
	Títol: Guia d'ús del Gestor d'Informació del CEE Jeroni de Moragas Autor: Carles Torras Font - Data: 20 d'abril de 2015	
	Revisió: 1 Pàgina: 9 de 10	


Benvingut al gestor d'informació del Jeroni de Moragas!

Catàleg de fotografies

Filtre

Filtrat per aula: Eclipsi, 2010-2011

Filtrat per calendari: Des de: 2015-05-05 Fins:


Filtrat per etiqueta:

Filtrat per alumne: Alba Fern M

Filtra

May 2015

Su	Mo	Tu	We	Th	Fr	Sa
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23

	DOCUMENT INTERN	I003_200415_JEMOR
	Títol: Guia d'ús del Gestor d'Informació del CEE Jeroni de Moragas Autor: Carles Torras Font - Data: 20 d'abril de 2015	Revisió: 1
		Pàgina: 10 de 10

Dades creades fictícies pel test del gestor

A fi i efecte de provar el funcionament del gestor d'informació, s'han generat diverses relacions de dades fictícies per usar-lo. Per facilitar-ne l'ús, a continuació es llisten les dades (per exemple, per conèixer *a priori* quin és l'usuari amb rol de director/a o per conèixer algun nom d'usuari i clau per accedir al gestor).

Categories: Director/a, Educador/a, Mestre/a, Logopeda, Fisioterapeuta, Administrador/a.

Municipis: Tots els 947 municipis de Catalunya amb la província on corresponen.

Professional (NIF, tutoria, nom, cognom 1, cognom 2, clau i nom de la categoria):

- '00000000A','no',,'Joana','Vilatorrada','Mer',a,'Administrador/a'
- '10000000A','no','Aurembiaix','Martínez','Olesa',a,'Director/a'
- '20000000A','si','Joan','Carrasco','Foix',a,'Mestre/a'
- '30000000A','no','Manel','Formosa','Vilella',a,'Mestre/a'
- '40000000A','no','Marcel.la','Vivancos','Peña',a,'Educador/a'
- '50000000A','si','Queralt','Isona','Salvadó',a,'Mestre/a'

Aules: Quart Creixent, Lluna Nova A, Lluna Nova B, Eclipsi, Lluna Plena, Logopèdia

Cursos: del 2010/11 al 2029/30.

Alumnes (nom, cognom 1, cognom 2, NIF i any d'entrada):

- 'Joan', 'Pere', 'Benet', '69767809E', '2003'
- 'Gili', 'Flix', 'Pered', '83976345H', '2001'
- 'Miki', 'Puig', 'Ferre', '45465564G', '2002'
- 'Joan', 'Albi', 'Pulla', '46571614M', '2005'
- 'Karl', 'Marx', 'Petra', '23466266H', '2006'
- 'Juli', 'Cesc', 'Paqui', '65156425B', '2007'
- 'Arla', 'Marc', 'Perri', '77676765G', '2008'
- 'Llum', 'Over', 'Kilmi', '71246146W', '2009'
- 'Alba', 'Ferm', 'Murts', '23565466J', '2012'
- 'Gila', 'Munt', 'Munts', '68554255H', '2011'

Notes:

- Alguns objectes no s'han introduït amb tota la informació (informació opcional). Tanmateix, tots els camps han estat provats.
- No s'ha introduït informació de tots els objectes. Només dels principals a partir dels quals es pot generar la resta. Per exemple, a partir d'un professional, aula i alumne, es pot generar una falta d'assistència.

A5.4.- I004. Guia d'ús dels ordinadors del CEE

	DOCUMENT INTERN	I004_280415_JEMOR
	Títol: Guia d'ús dels ordinador del CEE d'AJEM Autor: Carles Torras Font - Data: 28 d'abril de 2015	Revisió: 1
		Pàgina: 1 de 11

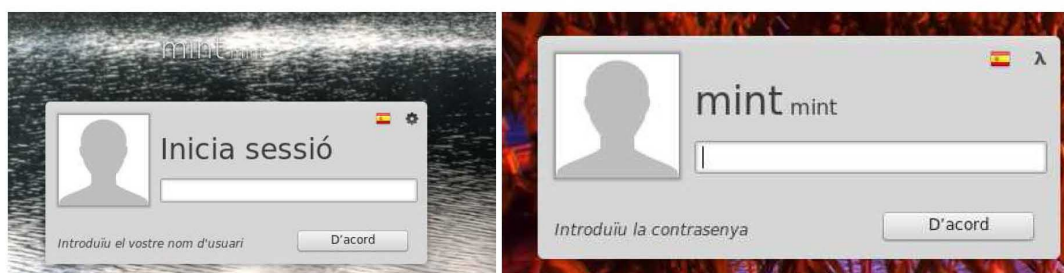
Guia d'ús dels ordinadors del Centre d'Educació Especial Jeroni de Moragas

Aquesta guia serveix per qualsevol ordinador del CEE d'AJEM, ja que tots ells contenen el mateix programari i configuració. Pretén ser una breu guia molt bàsica de les principals funcionalitats dels ordinadors a partir del programari instal·lat.

1.- Procediments bàsics	1
1.1.- Obrir un programa	3
1.2.- Obrir un arxiu	3
2.- Usuaris i permisos en Linux Mint i sistemes GNU/Linux	5
3.- Instal·lació de nous programes	7
4.- Processador de textos: LibreOffice Writer	8
4.1.- Creació de formularis amb LibreOffice Writer	9
5.- Documents adjunts	11
6.- Referències	11

1.- Procediments bàsics

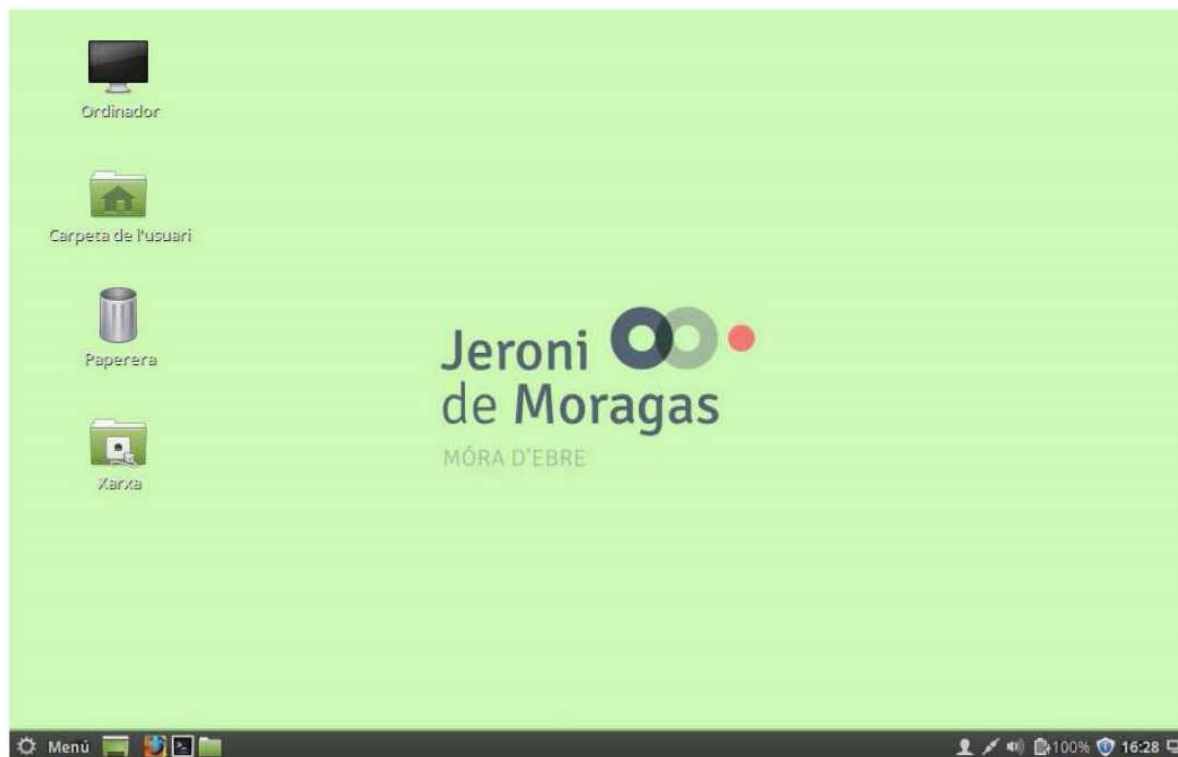
- En primer lloc, cal assegurar que l'ordinador està endollat, així com el maquinari complementari (per exemple, les impressores). A continuació, es pot posar en marxa l'ordinador mitjançant l'interruptor. De la mateixa manera, cal posar en marxa la resta de maquinari si es necessita.
- Després d'uns instants, **Linux Mint** (el sistema operatiu) [1] es posarà en marxa i ens mostrarà la següent pantalla d'inici de sessió:



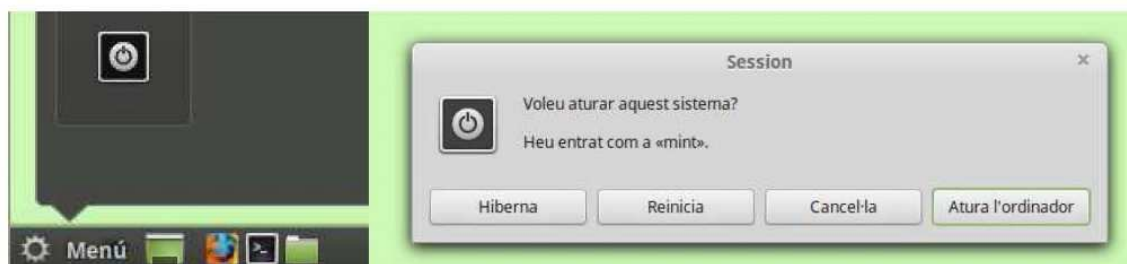
En funció de com s'hagi aturat l'ordinador la darrera vegada, pot ser que el sistema demani un nom d'usuari (com en el cas de la imatge de l'esquerra) o ja tingui un nom d'usuari introduït i només calgui introduir la clau d'accés (imatge de la dreta). En cas que hi hagi un nom d'usuari introduït i no sigui el desitjat, es pot prémer la tecla "Esc" per poder introduir el nom d'un altre usuari. Cal **no confondre** el nom d'usuari introduït amb la llista de noms d'usuari que surt en blanc, a sobre del diàleg d'accés.

	DOCUMENT INTERN	I004_280415_JEMOR
	Títol: Guia d'ús dels ordinadors del CEE d'AJEM Autor: Carles Torras Font - Data: 28 d'abril de 2015	Revisió: 1
		Pàgina: 2 de 11

- Un cop validat el nom d'usuari i contrasenya, es visualitzarà l'escriptori de l'usuari:



- Per **aturar l'ordinador**, des de l'escriptori, es fa clic a "**Menú**" i al darrer icona de la columna de l'esquerra que es titula "**Sortir**". Apareixerà un quadre de diàleg amb diverses opcions, de les quals s'ha de triar "Atura l'ordinador".

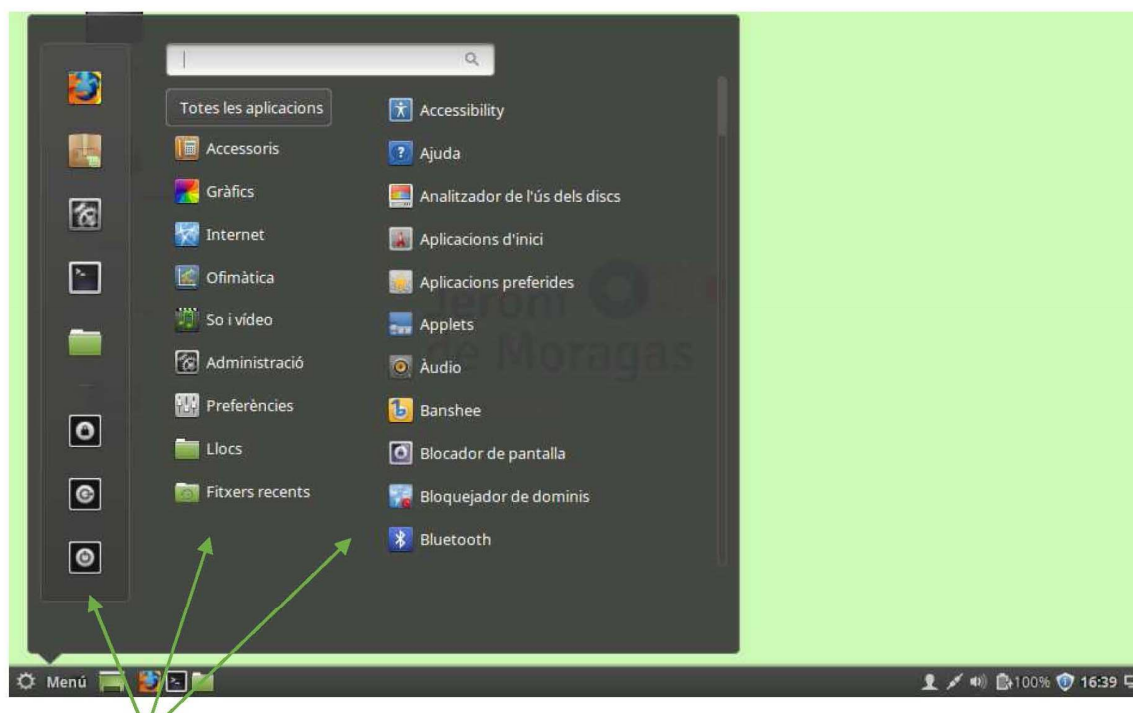


En funció del què es vulgui realitzar, existeixen diverses opcions. Per exemple, es pot desitjar **obrir un programa** o **obrir un arxiu**.

	DOCUMENT INTERN	I004_280415_JEMOR
	Títol: Guia d'ús dels ordinadors del CEE d'AJEM Autor: Carles Torras Font - Data: 28 d'abril de 2015	Revisió: 1
		Pàgina: 3 de 11

1.1.- Obrir un programa

S'accedeix als **programes** a través del "Menú":



Es mostren tres columnes. A la de l'**esquerra** es mostren **accessos directes** (per exemple, el primer és al navegador Firefox) i els darrers tres corresponen a opcions de tancament: bloquejar l'ordinador, tancar la sessió i aturar l'ordinador. A la columna del **mig** es mostren les **agrupacions de programa**: si es clica una opció, es mostren els **programes** d'aquella agrupació a la columna de la **dreta**.

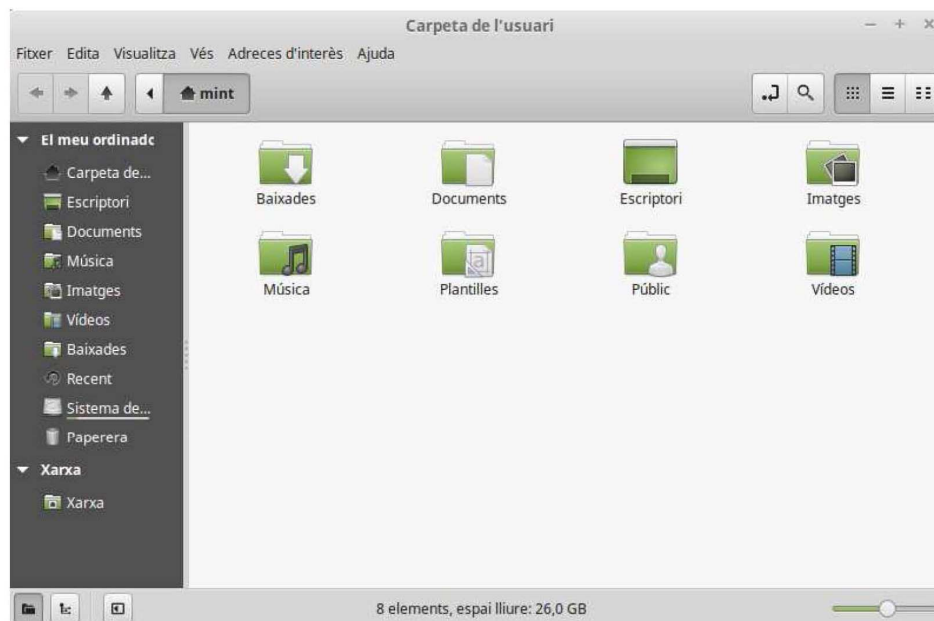
Principals programes:

- *Processador de textos*: Menú → Ofimàtica → LibreOffice Writer [2]
- *Calculadora*: Menú → Accessoris → Calculadora
- Tractament de so: Menú → So → Audacity [3]
- Tractament d'imatges: Menú → Gràfics → Editor d'imatges GIMP [4]
- Reproductor multimèdia: Menú → So → Reproductor multimèdia VNC

1.2.- Obrir un arxiu

Per aquesta opció, caldrà anar a la carpeta on estigui ubicat. Si està desat a la carpeta de l'usuari (opció recomanable per desar els arxius per tal de protegir-ne la privadesa, com s'explica en la secció següent), es pot fer doble clic directament damunt la "Carpeta de l'usuari". Es mostren les diverses carpetes de l'usuari classificades per continguts: imatges, documents, vídeos, baixades d'Internet, etc.

	DOCUMENT INTERN	I004_280415_JEMOR
	Títol: Guia d'ús dels ordinadors del CEE d'AJEM Autor: Carles Torras Font - Data: 28 d'abril de 2015	Revisió: 1
		Pàgina: 4 de 11



Una altra opció és fer doble clic a “Ordinador” i es mostrarà totes les unitats de disc dur, unitats de CD/DVD, unitats USB, et. que hi ha hagut:



Si es vol **recuperar algun arxiu esborrat**, des de qualsevol opció anterior, es pot accedir a la “**Paperera**”. Un cop clicada, es mostren els arxius a recuperar. Per fer-ho, cal col·locar-se damunt de l’arxiu i amb el botó dret del ratolí, clic a “**Restaurar**”. També es poden seleccionar i fer clic al botó superior “**Restaura els elements seleccionats**”. Seran desats a la carpeta on eren. De tant en tant és convenient buidar la paperera per no omplir innecessàriament el disc dur.

	DOCUMENT INTERN	1004_280415_JEMOR
	Títol: Guia d'ús dels ordinadors del CEE d'AJEM Autor: Carles Torras Font - Data: 28 d'abril de 2015	Revisió: 1
		Pàgina: 5 de 11



2.- Usuaris i permisos en Linux Mint i sistemes GNU/Linux

El sistema Windows, del qual la majoria d'usuaris estant acostumats, va ser un sistema pensat i dissenyat per un ús personal (no multiusuari) i local (no en xarxa). Una de les conseqüències principals i històriques és que l'usuari, en gran part, controla tots els arxius que conté el sistema i té un domini del funcionament del mateix (encara que no en sigui conscient). Aquest disseny és òptim quan les condicions d'ús personal i local existeixen.

Tanmateix, en molts casos (i el del CEE n'és un), l'ús és multiusuari i els ordinadors estant connectats en xarxa.

GNU / Linux, en ser un sistema dissenyat fonamentalment per a treball en xarxa, la seguretat de la informació que emmagatzemem és fonamental, ja que molts usuaris tindran o poden tenir accés a part dels recursos de programari (tant aplicacions com informació) i maquinari que estan gestionats en aquests ordinadors. Així, el sistema es va dissenyar des de bon començament amb una organització d'usuaris i permisos que garanteix l'accés controlat a la informació (i per tant arxius) que existeixen en el sistema. I alhora, als propis arxius del sistema, fet que augmenta la robustesa del mateix: no tots els usuaris poden modificar els arxius del sistema, poden introduir i executar virus, etc.

En GNU / Linux, els permisos o drets que els usuaris poden tenir sobre determinats arxius continguts en el sistema s'estableixen en tres nivells clarament diferenciats. Aquests són els següents:

- Permisos del propietari.
- Drets dels grups.
- Permisos de la resta d'usuaris (o també anomenats "els altres").

Per tenir clars aquests conceptes, en els sistemes en xarxa sempre hi ha la figura de l'administrador o *root*. Aquest administrador és l'encarregat de crear i donar de baixa usuaris, així com també, d'establir els privilegis que cada un d'ells tindrà en el sistema. Aquests privilegis s'estableixen tant per al directori HOME de cada usuari com per als directoris i arxius a què l'administrador decideixi que l'usuari pugui accedir.

	DOCUMENT INTERN	I004_280415_JEMOR
	Títol: Guia d'ús dels ordinadors del CEE d'AJEM Autor: Carles Torras Font - Data: 28 d'abril de 2015	Revisió: 1
		Pàgina: 6 de 11

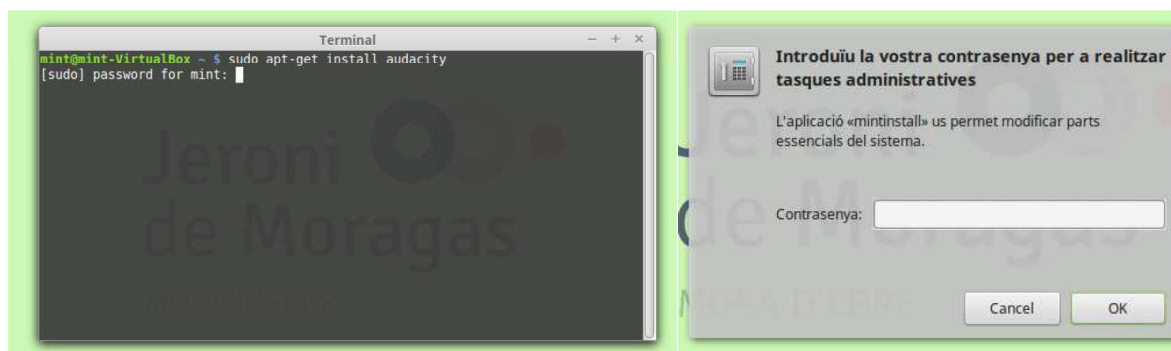
Permisos del propietari. El propietari és aquell usuari que genera o crea un arxiu / carpeta dins del seu directori de treball (HOME), o en algun altre directori sobre el qual tingui drets. Cada usuari té la potestat de crear, per defecte, els arxius que vulgui dins del seu directori de treball. En principi, ell i només ell serà el que tingui accés a la informació continguda en els arxius i directoris que hi ha en el seu directori HOME.

Drets dels grups. El més normal és que cada usuari pertanyi a un grup de treball. D'aquesta manera, quan es gestiona un grup, es gestionen tots els usuaris que pertanyen a aquest. És a dir, és més fàcil integrar diversos usuaris en un grup al qual se li concedeixen determinats privilegis en el sistema, que assignar els privilegis de forma independent a cada usuari.

Permisos de la resta d'usuaris. Finalment, també els privilegis dels arxius continguts en qualsevol directori, poden tenir-los altres usuaris que no pertanyin al grup de treball en què està integrat l'arxiu en qüestió. És a dir, als usuaris que no pertanyen al grup de treball en què està l'arxiu, però que pertanyen a altres grups de treball, se'ls denomina resta d'usuaris del sistema.

A **nivell d'usuari**, es poden destacar dues conseqüències d'això:

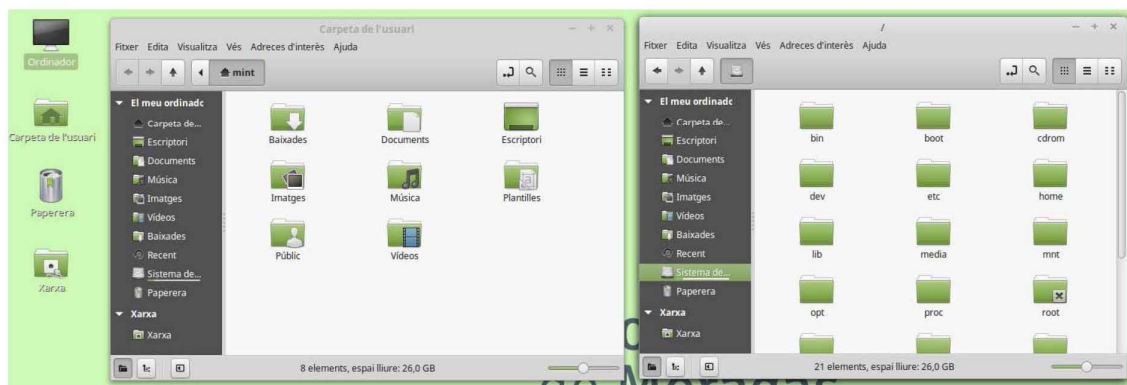
- Tots els arxius del sistema pertanyen a l'usuari administrador o *root*. Per tant, per alterar els arxius de sistemes cal estar registrat com a tal. Si no s'hi està, no és possible comprometre el sistema. Això també afecta als virus que veuen minvat el seu potencial, ja que en registre d'un usuari normal, tenen poca capacitat de causar malícies al sistema.
 - o Una conseqüència d'això, per exemple, és que si es vol instal·lar un programa nou, cal fer-ho com a usuari administrador. El "canvi" es pot fer des d'una terminal o simplement a nivell gràfic, introduint la clau de l'usuari administrador que el propi sistema demana automàticament. **Cal màxima cura en reconèixer si és adequat permetre l'operació i proporcionar la clau.**



En ambdós casos es pretén instal·lar l'aplicació "audacity". Mitjançant un terminal (figura esquerra) s'executa l'ordre "apt-get install" que significa "aconsegueix el paquet d'instal·lació dels repositoris en xarxa del sistema i instal·la'l". I "sudo" és l'ordre que indica fer-ho com a administrador (i per això, en primer lloc, es demana la clau). En la figura de la dreta es realitza el mateix però gràficament. I en aquest cas, només es demana directament la clau.

- Tots els arxius que crea un usuari s'emmagatzemen a la seva carpeta d'usuari HOME (i subcarpetes). Si s'accedeix al sistema amb un altre nom d'usuari, no es pot accedir a la informació i arxius del primer. Per tant, es garanteix la confidencialitat i seguretat.

	DOCUMENT INTERN	1004_280415_JEMOR
	Títol: Guia d'ús dels ordinador del CEE d'AJEM Autor: Carles Torras Font - Data: 28 d'abril de 2015	Revisió: 1
		Pàgina: 7 de 11



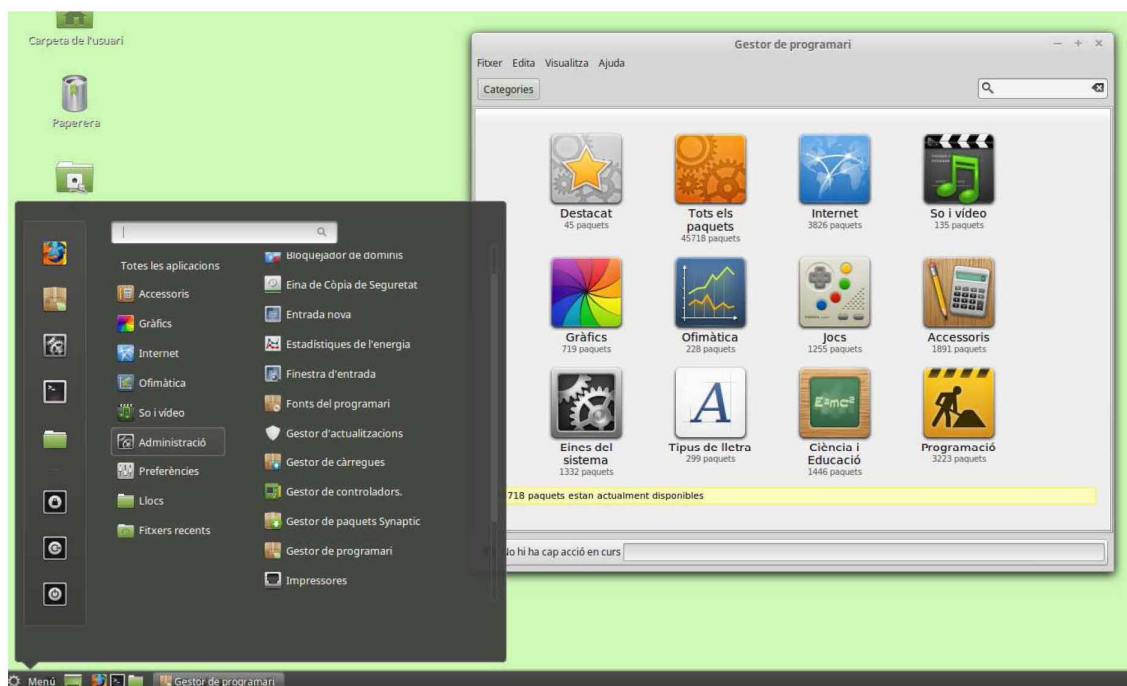
En les icones de l'escriptori es pot visualitzar la de "Carpeta d'usuari". Aquesta és la finestra del mig. És l'anomenada "HOME". Dins del sistema d'arxius globals (carpetes ubicades a l'arrel del disc dur), la carpeta HOME és un directori d'entre els diversos que formen part del sistema i que (la resta) són propietat de l'usuari administrador (finestra de la dreta).

Per més informació sobre el tema, es pot consultar el següent bloc, el qual descriu el tema amb un llenguatge no tècnic: [5].

3.- Instal·lació de nous programes

Això es pot realitzar mitjançant terminal i gràficament (com s'ha mostrat en les anteriors figures). A nivell d'usuari bàsic es recomana realitzar-ho gràficament:

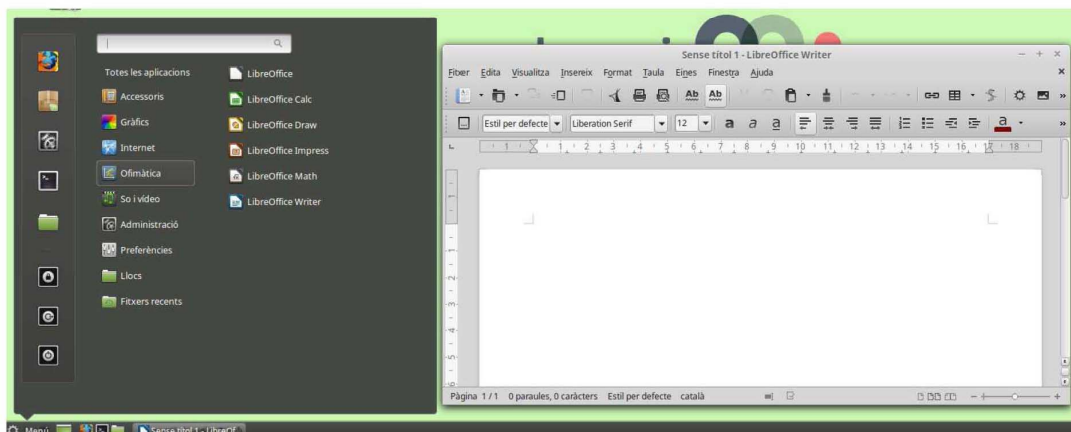
- Obrir el "Gestor de programari" (Menú → Administració → Gestor de Programari). Es demanarà la clau de l'administrador.
- Cerca el programa a instal·lar i executar-ho.



	DOCUMENT INTERN	I004_280415_JEMOR
	Títol: Guia d'ús dels ordinador del CEE d'AJEM Autor: Carles Torras Font - Data: 28 d'abril de 2015	Revisió: 1

4.- Processador de textos: LibreOffice Writer

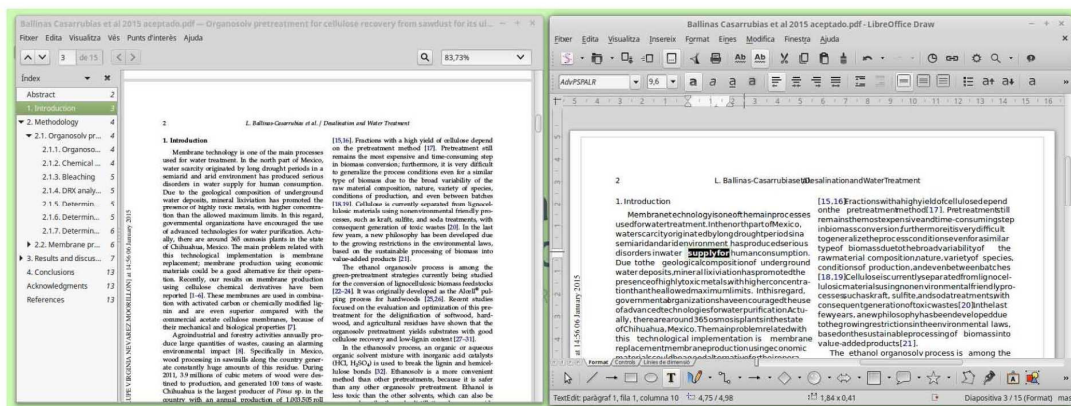
Per obrir el processador de textos cal anar a Menú → Ofimàtica → LibreOffice Writer.



El funcionament és molt semblant a altres processador de textos (veure la guia d'ús indicada a sota).

El format per defecte és el ODT (*Open Document Format*). És el format estàndard obert dels documents de processadors de textos (PT). La majoria de PT's són capaços de tractar amb aquest format. Tanmateix, LibreOffice Writer també pot tractar amb altres formats com el DOC/DOCX de Microsoft Word o amb PDF (*Portable Document Format*), tant per generar documents com per obrir-ne. Per obrir (importar) un document PDF s'ha de realitzar com es faria amb qualsevol altre (obrir normal).

S'ha de considerar que com en tots els programes (també en Microsoft Word) el resultat de la importació del PDF pot suposar pèrdues de fidelitat respecte el document original. I s'ha de considerar que serà sobretot efectiva si el document PDF conté informació provinent de text i no d'imatge. Per exemple, suposem que es té un document PDF amb text. Aquest text pot provenir d'un editor de textos o d'una pàgina escanejada. Si és la primera opció, la importació del PDF serà exitosa. Però si és la segona opció (text provinent d'una pàgina escanejada i per tant, amb format d'imatge), el text no es recuperarà. Per aquests casos, cal un programa d'OCR (*Optical Character Recognition*). Aquests són programes capaços de desxifrar lletres a partir de les seves formes (per exemple, existeix un eina online gratuïta per fer-ho: <http://www.free-ocr.com/>).



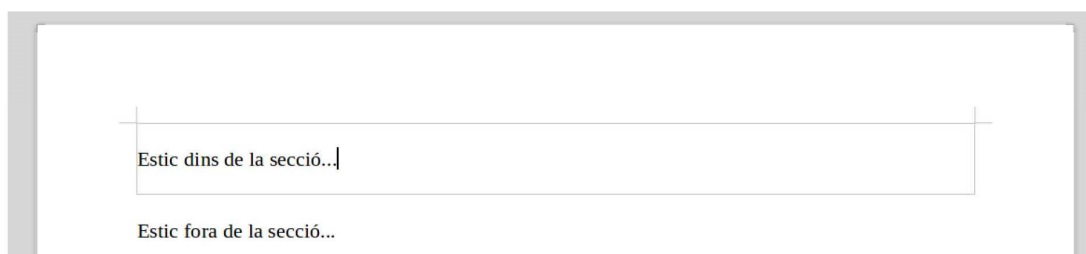
Exemple d'un arxiu PDF obert amb un programa lector de PDF's (esquerra) i importat al LibreOffice Writer (dreta). En aquest darrer, l'arxiu es pot editar completament.


	DOCUMENT INTERN	I004_280415_JEMOR
	Títol: Guia d'ús dels ordinador del CEE d'AJEM Autor: Carles Torras Font - Data: 28 d'abril de 2015	Revisió: 1
		Pàgina: 9 de 11

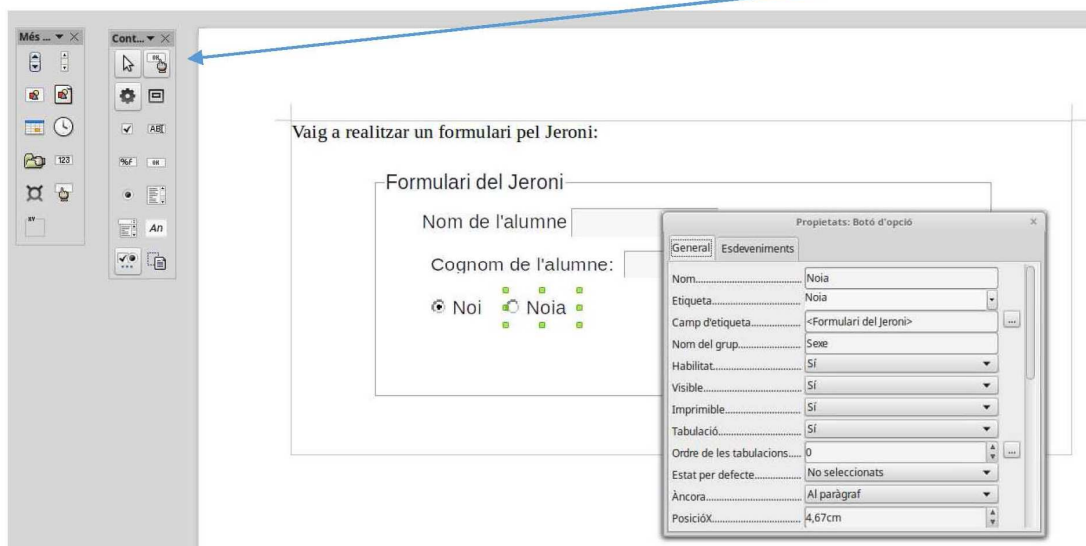
4.1.- Creació de formularis amb *LibreOffice Writer*

A continuació s'exemplifica com crear un formulari. Aquest es protegirà de manera que només es puguin omplir els camps.

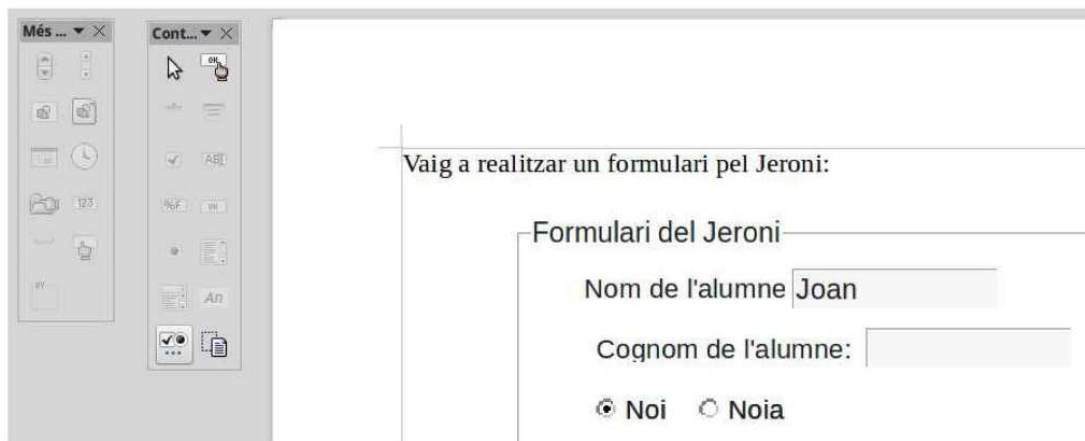
- Obrir el LibreOffice Writer i crear un nou document (per defecte, això darrer ja succeeix).
- Crear una nova secció que contindrà el formulari: menú "insereix" → "secció". S'obrirà un quadre de diàleg del qual només haurem de clicar al botó de sota "insereix". Es visualitzarà en el document que s'ha creat un quadre corresponent a la secció. Caldrà editar el formulari dins d'aquest requadre (que es pot ampliar tant com faci falta):



- Obrir la barra d'eines de formularis: menú "visualitza" → "barres d'eines" → "controls de formulari".
- Afegir al document els camps desitjats (seleccionar i arrastrar). Per cada element que s'insereix, fent doble clic sobre el mateix, s'obre el quadre de diàleg de configuració. Per exemple, pels *radiobutton*, si es vol crear un grup de dos (com el de l'exemple) on seleccionar-ne un implica que l'altre es de-seleccioni, cal que ambdós tinguin el mateix nom de grup.
- Clicant sobre qualsevol camp amb el botó dret del ratolí, es mostren altres opcions, com organitzar els elements (per exemple, enviar el panell al fons i que els elements quedin davant).
- Quan es disposa un formulari en un document, existeixen dos modes: el de disseny del formulari i el d'usuari. Mentre es dissenya, cal tenir activat el primer (per defecte) i quan s'ha dissenyat, cal activar el segon (no es podrà editar el formulari i es podrà usar). Per estar en mode disseny cal tenir clicat el botó  i per estar en mode usuari cal que estigui no clicat. Aquest botó es troba a la barra d'eines de formulari (el de dalt de tot a la dreta).



	DOCUMENT INTERN	I004_280415_JEMOR
	Títol: Guia d'ús dels ordinador del CEE d'AJEM Autor: Carles Torras Font - Data: 28 d'abril de 2015	Revisió: 1
		Pàgina: 10 de 11



Vaig a realitzar un formulari pel Jeroni:

Formulari del Jeroni

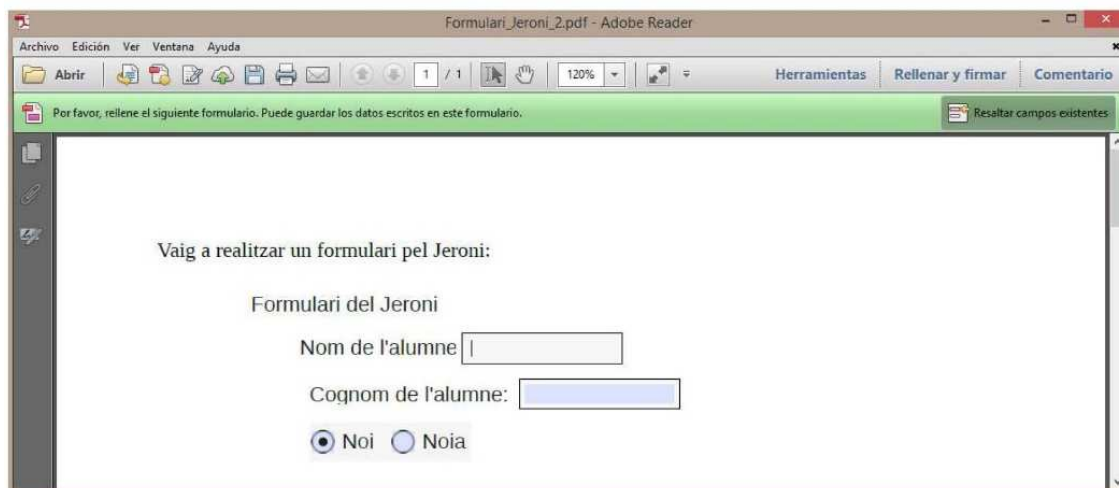
Nom de l'alumne Joan

Cognom de l'alumne:

Noi Noia

Disseny d'un formulari. La primera figura mostra la construcció del formulari. El botó de disseny està clicat i per tant es poden editar els camps. La figura de sota mostra l'estat d'usuari (no es pot dissenyar el formulari però es pot usar). El botó de disseny no està clicat i la resta d'icones romanen desactivades.

- Finalment el sistema permet protegir el document per tal que no es pugui editar el contingut i només es pugui omplir la informació del formulari. Per fer-ho:
 - Menú "format" → "seccions" i de la secció a la qual s'hagi editat el formulari, clicar a "Protegit" i "Amb contrasenya" (si es vol) de l'apartat "Protecció contra escriptura".
 - Si es selecciona el darrer, caldrà introduir la contrasenya i verificar-la. A partir d'aquí ja no es podrà editar dins la secció, només completar els camps del formulari.
- Desar l'arxiu.
- Si posteriorment s'omple el formulari i es torna a desar l'arxiu, es desarà amb el formulari ple.
- Si es vol tornar a editar la secció, caldrà desactivar l'opció de protecció (inversament a com s'ha fet per activar-la).
- LibreOffice Writer* també permet desar el document com a PDF. I llavors, el formulari es podrà també omplir com a PDF (però en funció del programa que s'utilitzi pot no permetre desar el PDF amb les dades omplertes, només imprimir-lo amb elles).
 - Menú "fitxer" → "exporta com a PDF" i assegurar-se que l'opció "crea un formulari PDF" està activada.



Formulari_Jeroni_2.pdf - Adobe Reader

Archivo Edición Ver Ventana Ayuda

Abrir Herramientas Rellenar y firmar Comentario

Por favor, rellene el siguiente formulario. Puede guardar los datos escritos en este formulario. Resaltar campos existentes

Vaig a realitzar un formulari pel Jeroni:

Formulari del Jeroni

Nom de l'alumne

Cognom de l'alumne:

Noi Noia

	DOCUMENT INTERN	I004_280415_JEMOR
	Títol: Guia d'ús dels ordinadors del CEE d'AJEM Autor: Carles Torras Font - Data: 28 d'abril de 2015	Revisió: 1
		Pàgina: 11 de 11

5.- Documents adjunts

Aquesta guia és molt breu i només introductòria. Amb ella s'adjunten els següents documents que són guies més complertes dels diversos programes:

- Guia de l'usuari de **Linux Mint**.
- Guia de l'usuari de **LibreOffice**.
- Guia de l'usuari de **Gimp**.
- Guia de l'usuari d'**Audacity**.

6.- Referències

1. Mint, L. *Distribució Linux Mint*. 2015; Available from: <http://www.linuxmint.com/>.
2. LibreOffice. *Projecte LibreOffice*. 2015; Available from: <https://ca.libreoffice.org/>.
3. d'Audacity, C.d.d. *Projecte Audacity*. 2015; Available from: <http://audacity.sourceforge.net/>.
4. GIMP, C.d.d.d. *GIMP. The GNU Image Manipulation Program*. 2015; Available from: <http://www.gimp.org/>.
5. Perseo, *Bloc de Perseo*. 2012. <http://blog.desdelinux.net/permisos-y-derechos-en-linux/>