



### **Creative Commons**

Aquest treball està subjecte -excepte que s'indiqui el contrari- a una llicència de *Reconeixement-NoComercial-CompatirIgual 3.0* de Creative Commons. Podeu copiar, distribuir, comunicar públicament l'obra i fer-ne obres derivades, sempre i quan es compleixin les condicions de reconeixement i compartir igual.

La llicència completa es pot consultar a <http://creativecommons.org/licenses/by-nc-sa/3.0/es/deed.ca>

## 1. Resum

El present treball de fi de Màster té com a objectiu la creació d'una aplicació de gestió centralitzada d'accessos d'usuaris a servidors GNU/Linux, de manera que els autoritzats puguin accedir als servidors corresponents sense necessitat d'utilitzar una validació per *username* i *contrasenya*; aquest punt serà possible gràcies a la validació per clau pública sota protocol SSH.

L'actual projecte es basa en la creació d'una aplicació web basada en programari lliure, en la qual s'hi apliquen els coneixements adquirits al llarg del Màster en Programari Lliure de la UOC. Assignatures com *Desenvolupament d'aplicacions web*, *Implantació de sistemes de programari lliure*, *Administració de sistemes GNU/Linux*, i *Bases de dades*, han permès obtenir la base d'un producte fonamentat sobre programari lliure. Així, el treball de fi de Màster respon a una necessitat real d'empresa, permetent una reducció de temps en tasques repetitives i, alhora, una reducció real de costos i un millor aprofitament de recursos.

A més, s'ha aprofitat el projecte per aplicar el model MVC a una solució web implementada en PHP. Així, a part de realitzar els passos comuns a tot projecte (*fases d'estudi de viabilitat*, *d'anàlisi*, *de disseny* i *de desenvolupament*), ha estat necessari aprendre el llenguatge de programació PHP i cercar i estudiar l'opció més adient per implementar l'aplicació seguint el paradigma MVC.

En base al punt de sortida indicat, finalment s'ha escollit el *framework* TinyMVC: malgrat sigui una solució limitada d'implementació del model MVC sobre PHP, la seva senzillesa ha permès assolir els objectius d'agilitat, ràpida corba d'aprenentatge, facilitat de manteniment i transparència d'implementació amb un cost ínfim d'espai en disc i recursos de sistema.

## Índex de continguts

1. Resum.....	3
2. Introducció.....	7
2.1. Escenari.....	7
2.2. Objectius .....	7
2.3. Components de programari lliure utilitzats.....	7
2.4. Planificació.....	8
2.5. Problemes i solucions.....	9
3. Estudi de viabilitat.....	10
3.1. Descripció general i abast del projecte.....	10
3.2. Situació actual del sistema.....	11
3.3. Requisits a complir per part de la solució escollida.....	13
3.4. Descripció de les solucions considerades.....	14
3.5. Anàlisi de costos/beneficis i riscos de les diferents solucions.....	14
3.6. Selecció de la solució.....	15
4. Anàlisi.....	16
4.1. Definició del sistema.....	16
4.2. Establiment de requisits.....	17
4.3. Identificació d'interfícies d'usuari.....	28
4.4. Pla de proves.....	29
5. Disseny.....	30
5.1. Definició de l'arquitectura.....	30
5.2. Concreció dels subsistemes.....	30
5.3. Arquitectura del sistema d'aplicació.....	32
5.4. Diagrames de casos d'ús.....	33
5.5. Diagrames addicionals.....	34
5.6. Diagrama de classes.....	38
5.7. Diagrama ER.....	38
5.8. Disseny de la capa de persistència.....	39
5.9. Creació de la interfície gràfica.....	40
6. Desenvolupament.....	49
6.1. Estructura.....	49
6.2. Desplegament.....	52
7. Conclusions.....	53
7.1. Globals.....	53
7.2. Aplicació.....	53
8. Glossari.....	54
9. Bibliografia.....	55
10. Annexes.....	56
10.1. Sistema de logs.....	56
10.2. Script SQL de creació de la DB MySQL.....	57

## Índex de figures

Figura 1: escenari global d'accés.....	10
Figura 2: subsistemes de l'aplicació web del projecte.....	12
Figura 3: cas d'ús del subsistema d'identificació.....	17
Figura 4: casos d'ús del subsistema de gestió d'usuaris.....	18
Figura 5: casos d'ús del subsistema de gestió de servidors.....	21
Figura 6: casos d'ús del subsistema de gestió d'accessos.....	25
Figura 7: interfície corresponent al menú inicial de l'aplicació.....	28
Figura 8: interfície comuna a tots els casos d'ús.....	28
Figura 9: comunicacions establertes entre sistemes.....	30
Figura 10: comunicacions establertes entre subsistemes.....	31
Figura 11: exemple d'operació d'actualització executada pel model MVC de TinyMVC.....	33
Figura 12: cas d'ús del subsistema d'identificació.....	33
Figura 13: diagrama de seqüència del cas d'ús identificar empleat, amb validació correcta.....	34
Figura 14: diagrama de seqüència del cas d'ús identificar empleat, amb validació incorrecta.....	35
Figura 15: diagrama de seqüència del cas d'ús canvi contrasenya, amb validació correcta.....	35
Figura 16: diagrama de seqüència del cas d'ús crear usuari.....	36
Figura 17: diagrama de seqüència del cas d'ús modificar servidor.....	36
Figura 18: diagrama de seqüència del cas d'ús esborrar grup de servidors.....	37
Figura 19: diagrama de seqüència del cas d'ús crear accés.....	37
Figura 20: diagrama de classes, escenari general.....	38
Figura 21: diagrama ER, escenari general.....	38
Figura 22: vista d'identificació d'usuari.....	42
Figura 23: vista d'identificació incorrecta d'usuari.....	43
Figura 24: vista de canvi de contrasenya.....	43
Figura 25: vista del menú principal.....	43
Figura 26: vista d'alta d'usuari.....	44
Figura 27: missatges informatius comuns a totes les vistes.....	45
Figura 28: vista d'usuaris.....	45
Figura 29: vista de consulta d'usuaris.....	45
Figura 30: vista d'alta de grup d'usuaris.....	46
Figura 31: vista de grups d'usuaris.....	46
Figura 32: vista de consulta d'usuari.....	46
Figura 33: vista d'alta de servidors.....	47
Figura 34: vista de consulta de servidors.....	47

---

Figura 35: vista de consulta de servidors filtrada.....	47
Figura 36: vista d'alta d'accés.....	48
Figura 37: vista de consulta d'accessos.....	48
Figura 38: estructura de carpetes del framework TinyMVC.....	49
Figura 39: arxius de configuració del framework TinyMVC.....	49
Figura 40: configuració d'accés a DB del framework TinyMVC.....	49
Figura 41: controladors definits al framework TinyMVC.....	50
Figura 42: models definits al framework TinyMVC.....	50
Figura 43: vistes definides al framework TinyMVC.....	51
Figura 44: vista inicial de l'aplicació desplegada.....	52

## 2. Introducció

### 2.1. Escenari

En la situació econòmica actual, pren especial importància qualsevol solució que permeti un augment en l'eficiència d'una empresa. Dins de l'àmbit de l'administració de sistemes informàtics, una possible via per obtenir aquesta millora és reduir el temps invertit en fer tasques habituals i repetitives. El present projecte pretén donar resposta al procés de connexió als servidors.

Els administradors i personal autoritzat connecten a diari amb els servidors que administren. Tenint en compte que un gran nombre de servidors dels CPD utilitzen sistemes UNIX i GNU/Linux, les connexions es realitzen sobre protocol SSH mitjançant la identificació amb usuari i contrasenya. Aquest fet implica costos temporals en la identificació d'accés a cada servidor, en la gestió de contrasenyes d'usuaris individuals i/o compartits, en la gestió de seguretat de les identifikacions, etc.

L'actual projecte pretén crear una aplicació web que permeti als administradors poder accedir a llurs servidors sense necessitat de validar-se amb usuari i contrasenya, mitjançant l'ús de validació per criptografia de clau pública.

### 2.2. Objectius

Es defineixen tres objectius principals pel PFM:

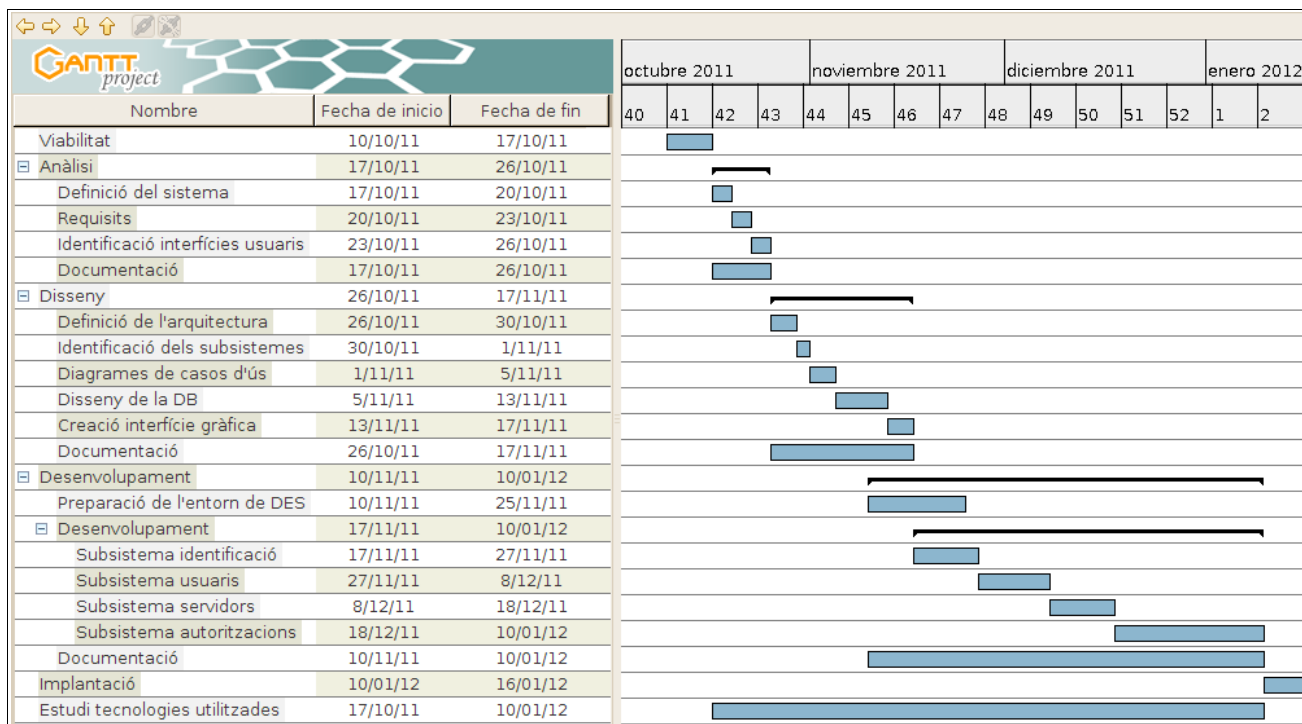
- Aprendre les tecnologies necessàries per implementar el projecte: serà un punt fonamental per poder implementar el PFM, i més tenint en compte que els coneixements inicials d'aspectes com PHP i/o frameworks MVC són limitats.
- Desenvolupar els quatre subsistemes de l'aplicació: *identificació*, *gestió d'usuaris*, *gestió de servidors* i *gestió d'accessos*. La base mínima a implementar acordada amb l'empresa serà la persistència de dades de cadascun dels subsistemes i, si es disposa de temps, s'aniran ampliant funcionalitats i subsistemes, com el de *comunicació* i el d'*administració*.
- Generar la documentació corresponent per totes les fases que conté el PFM: no únicament la relativa a les entregues parcials i final del projecte, sinó també documentació de tipus interna de l'aplicació (comentaris a nivell de codificació).

### 2.3. Components de programari lliure utilitzats

Programari	Tipus	Ús
Ubuntu	Sistema operatiu	Sistema operatiu utilitzat pel desenvolupament de l'aplicació, i generació de tota la documentació.
Apache HTTPd	Servidor web	Servidor web on es desplega i s'executa el projecte.
PHP	Servidor PHP	Llenguatge de programació utilitzat per la codificació de l'aplicació.

Programari	Tipus	Ús
OpenOffice Writer	Processador de textos	Utilitzat per generar la documentació de totes les entregues del projecte, així com de la present Memòria.
Dia	Editor de diagrames	Programari amb el qual s'han creat tots els diagrames UML de la Memòria.
Eclipse for PHP developers	IDE	Entorn de desenvolupament del projecte.
MySQL	SGBD	Programari que implementa la capa de persistència de l'aplicació.
TinyMVC	Framework MVC per PHP	Framework que implementa el paradigma PHP i que s'utilitza per estructurar l'aplicació.
Gimp	Edició d'imatges	Programari de retoc d'imatges utilitzar per adequar certes imatges i icones que posteriorment s'han utilitzat a les vistes.
Firefox	Navegador web	Utilitzats per testejar i validar l'aplicació.
Chromium	Navegador web	
VirtualBox	Entorn de virtualització	Utilitzat per validar el desplegament de l'aplicació en un entorn completament net.
XVidCap	Capturador video	Programari utilitzar per capturar l'ús de l'aplicació i desar-ho com a arxiu de vídeo.

## 2.4. Planificació





## 2.5. Problemes i solucions

El principal problema detectat en la realització del projecte ha la gestió del temps per una malaltia familiar greu que ha coincidit en el mateix període de temps. Aquest punt ha provocat que la planificació inicial del punt 2.3 no s'hagi pogut seguir, i que no s'hagi pogut implementar cap funcionalitat addicional a les inicialment establertes.

Ha estat possible superar aquest impediment gràcies a la utilització d'un *framework* MVC amb una corba d'aprenentatge ràpida, a la *reutilització* de codi entre subsistemes (el disseny previ ha estat vital), i a l'ús de conceptes bàsics *agile*<sup>1</sup> de desenvolupament de programari. L'adaptació per part del consultor i l'empresa a aquesta situació també ha estat un factor determinant.

---

<sup>1</sup> S'han anat implementant cada funcionalitat de subsistema de forma completa i seqüencial a la resta de funcionalitats: fins que una opció no ha estat totalment operativa, després d'haver validat amb el client que compleix els requisits mínims, no es passa a realitzar la següent funcionalitat. Més informació a [http://es.wikipedia.org/wiki/Desarrollo\\_%C3%A1gil\\_de\\_software](http://es.wikipedia.org/wiki/Desarrollo_%C3%A1gil_de_software) i <http://knol.google.com/k/desarrollo-ágil-de-software>,

## 3. Estudi de viabilitat

### 3.1. Descripció general i abast del projecte

UPCnet, empresa en la qual s'ha desenvolupat el present projecte, és una empresa del grup UPC creada per la Universitat Politècnica de Catalunya per a la prestació de serveis en tots els àmbits de les TIC.

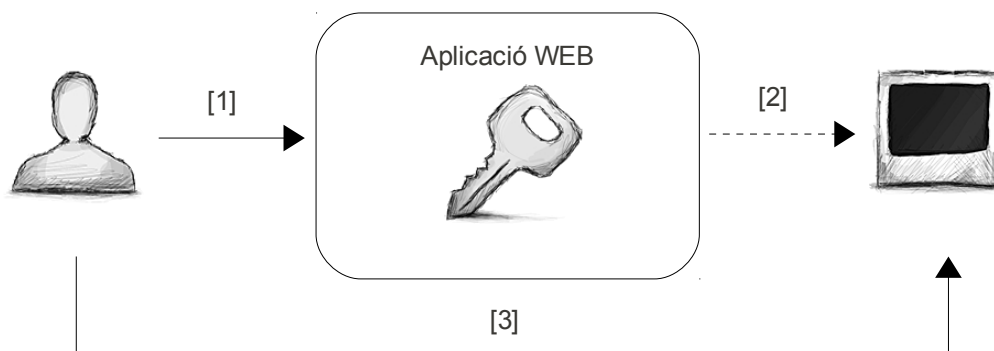
Un dels àmbits en el qual es realitza una major inversió de temps és el de l'administració de sistemes. Per aquest motiu, i tenint en compte la situació actual econòmica, s'ha de tenir en consideració qualsevol projecte que permeti una reducció de temps en la realització de tasques repetitives, fet que implica directament una reducció de costos.

Segons la definició disponible a la Wikipedia, un administrador de sistemes *és un professional dedicat al manteniment i funcionament d'un sistema informàtic i/o xarxa*; a tall d'exemple, algunes d'aquestes tasques de manteniment i funcionament són el suport a usuaris, la supervisió de sistemes, la resolució d'incidències, la instal·lació de productes, la creació d'scripts per facilitar les tasques diàries, etc.

El present projecte es fonamenta en tres pilars per assolir una reducció de costos en l'administració diària dels entorns gestionats:

- **Naturalesa dels servidors:** el parc de servidors administrat es caracteritza per pertànyer majoritàriament a la família de GNU/Linux (Red Hat, Ubuntu) i UNIX (IBM AIX, Sun Solaris).
- **Accés homogeni als servidors:** qualsevol tasca administrativa a realitzar sobre un servidor GNU/Linux i UNIX es fa mitjançant el protocol SSH, permetent establir connexions fiables i segures.
- **Accessos múltiples als servidors:** hi ha múltiples usuaris que requereixen d'accés al parc de servidors GNU/Linux i UNIX per poder realitzar les seves tasques. En funció del seu perfil, requeriran uns privilegis concrets d'accés.

En base a aquest escenari, l'objectiu principal del projecte és la creació d'una aplicació web de gestió d'accessos als servidors, la qual permeti als usuaris connectar amb els servidors amb uns determinats privilegis, i sense necessitat d'utilitzar una identificació amb contrasenya.



**Fig. 1 :** l'usuari es dona d'alta a l'aplicació de gestió d'accessos [1], la qual li habilita l'accés per SSH al servidor destí mitjançant validació per PKI [2], moment a partir del qual l'usuari podrà accedir al servidor destí de forma immediata, sense necessitat d'utilitzar contrasenya.

Des del punt de vista econòmic, perquè el projecte sigui viable serà precís poder-lo realitzar amb la menor despesa possible, ja que el cost no estava previst als pressupostos anuals de l'empresa.

Des del punt de vista tècnic, en ser un projecte basat en solucions obertes, l'enfocament coincideix amb la política de l'empresa d'utilitzar programari lliure sempre i quan sigui possible. Així mateix, el personal de què disposa està qualificat tècnicament per poder-lo gestionar posteriorment sense problemes.

Des del punt de vista legal, les solucions basades en programari lliure ofereixen molta flexibilitat, i permeten un grau de llibertat molt elevat a l'hora de continuar adaptant l'aplicació a les necessitats reals dels usuaris, així com de distribució del projecte a nous clients en cas de detectar la necessitat.

Des del punt de vista operatiu, el present projecte ha de poder oferir totes aquelles necessitats que la solució actual de l'empresa no contempla: la solució implementada és una aproximació parcial a les necessitats reals de l'entorn.

L'abast del projecte afectarà a dos àmbits concrets:

- *Infraestructura*: degut a la contenció de la despesa, s'utilitzarà el mateix servidor que actualment ofereix el servei de gestió d'accessos. Així s'aprofita un entorn basat en programari lliure amb visibilitat a nivell de xarxa de la resta de servidors, i s'eliminen els costos derivats de la creació del servidor, d'entrada a producció del mateix, i d'instal·lació del programari requerit.
- *Conscienciació*: s'haurà de fer una política de conscienciació de l'aplicació a tots els departaments de l'empresa susceptibles d'utilitzar-la, i destacar-ne els avantatges per tal que la reducció de costos no només sigui efectiva a nivell d'administració sinó també dins de l'àmbit de projectes. Serà decisiu informar i implicar els responsables de cadascun dels departaments afectats.

El projecte afectarà bàsicament al grup d'administradors de l'empresa, ja que requereixen d'accés als servidors com a usuaris amb privilegis elevats.

S'ha plantejat a l'empresa que altres grups puguin accedir als servidors amb privilegis menors, però finalment la necessitat ha quedat fora de l'abast del projecte. Malgrat tot, la solució plantejada ha tingut en compte aquesta casuística a l'hora de dissenyar l'aplicació, per tal que la seva implementació en un futur no impliqui un gran cost.

## 3.2. Situació actual del sistema

Amb l'objectiu de poder detectar les necessitats reals de l'entorn, i fer un diagnòstic tan complet com sigui possible, s'ha decidit treballar des d'un principi amb els administradors de l'aplicació web que actualment ofereix aquest servei. En ser qui més contacte directe i ús intensiu tenen amb l'aplicació actual, són qui de forma més clara coneixen les mancances de l'actual solució.

La situació actual és la següent:

- *Aplicació web*: en l'actualitat el sistema es basa en una arquitectura LAMP<sup>2</sup>, gestionat pel departament d'Infraestructura. El manteniment de l'aplicació es realitza mitjançant qualsevol editor disponible al SO (per exemple, vi o vim), i requereix un perfil tècnic amb coneixements d'HTML, PHP, Javascript, CSS i MySQL.

---

2 Acrònim de Linux, Apache, MySQL i PHP

El sistema web està estructurat en quatre grans àrees o subsistemes:

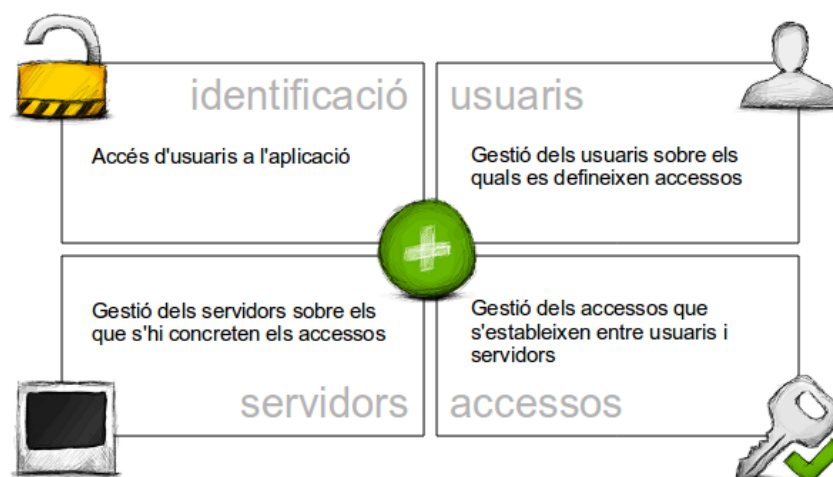




Fig. 2 : subsistemes en que està dividida l'aplicació web del projecte.












Una vegada analitzat el sistema actual, s'ha arribat a les següents conclusions:

- El conjunt del sistema web està orientat als administradors, permetent definir únicament accessos als servidors mitjançant superusuaris de SO (`root`); aquest enfocament impedeix que departaments de Projectes (desenvolupadors) no puguin utilitzar l'aplicació.
- La interfície web és en alguns punts confosa: es requereix d'un cert de temps per aprendre el funcionament de l'aplicació. A més, hauria de ser més agradable.
- La distribució interna de l'aplicació a nivell de programació és complicada de gestionar: qualsevol canvi que a realitzar necessita d'una inversió de temps per saber què s'ha de modificar i on. La documentació és ínfima.
- La gestió d'errors (excepcions) s'ha de millorar: els missatges informatius que retorna l'aplicació han de ser més descriptius.
- La interrelació entre grups d'usuaris i grups de servidors és millorable: algunes operatives puntuals no funcionen com seria desitjable.
- Les auditories de l'aplicació no estan implementades: per motius de seguretat, qualsevol canvi en la gestió d'accessos hauria de quedar reflectit a l'aplicació.

Per tots aquests motius, l'opció d'implementar una aplicació completament nova passa per davant de qualsevol millora que es vulgui fer de l'actual.

### 3.3. Requisits a complir per part de la solució escollida

A partir de les reunions realitzades amb els administradors de l'actual solució, s'han detectat les següents necessitats, classificades segons el nivell de prioritat alt , mig  i baix :

Tipus	Nivell	Requisit
Tècnic		Arquitectura del maquinari: el contingut de l'aplicació s'ha de poder servir des del servidor que actualment dona aquest servei. No ha de sobrepassar els seus recursos físics.
		Arquitectura del programari: ha d'utilitzar el paradigma LAMP. El sistema operatiu ha de ser GNU/Linux, el servidor web Apache HTTPd, la capa de persistència s'ha d'oferir amb MySQL, i PHP com a llenguatge de programació interpretat per la creació de pàgines web dinàmiques.
		Programació interna: ha de ser senzilla de seguir i de mantenir a posteriori. És imprescindible que el codi estigui ben documentat.
		Normatives i/o estàndards: el lloc web haurà de complir uns mínims per tal que no doni cap problema d'accessibilitat i operativitat amb el navegador Firefox (l'utilitzat a les imatges dels ordinadors de l'empresa).
		Auditories: les operacions executades per l'aplicació han de quedar registrades.
Operatiu		Look & feel: l'aplicació web ha de ser visualment atractiva, senzilla i fàcil d'utilitzar (instintiva). S'ha d'estructurar de forma clara en quatre blocs, corresponents als subsistemes d'identificació, usuaris, servidors i accessos.
		Gestió de l'aplicació: l'operativa l'ha de poder fer una persona que no tingui coneixements d'HTML, PHP, Javascript ni MySQL.
Legal		Llicència SO: el sistema operatiu ha d'utilitzar una llicència lliure i gratuïta. Per qüestions econòmiques (preparació d'un nou servidor, pagament pel servei d'actualitzacions de la distribució, etc), s'haurà d'utilitzar alguna distribució de GNU/Linux.
		Llicències de programari: hauran de ser poc restrictives, prioritàriament GPL.
Econòmic		Despeses nul·les a nivell de llicències de SO i de programari: únicament es contemplaran escenaris que no requereixin de pagament per obtenir una llicència determinada pel SO i/o pel programari utilitzat per executar l'aplicació (webserver, DB relacional, etc.)
		Reutilització: és de vital importància ser capaços de reutilitzar la infraestructura (servidor) que actualment ofereix aquest tipus de servei.

### 3.4. Descripció de les solucions considerades

En base als requisits detectats a l'apartat anterior, l'aplicació s'ha d'executar en un SO GNU/Linux, amb el que qualsevol opció basada en entorns Microsoft i/o UNIX queden automàticament descartades. El programari on sostenir l'aplicació web també està definit pels requisits: Apache HTTPd pel servidor de pàgines web, MySQL per la persistència de dades, i PHP com a llenguatge de programació interpretat per la creació de pàgines web dinàmiques.

Es valoren dues opcions basades en GNU/Linux:

- *Ubuntu Server LTS*<sup>3</sup>: les versions LTS es caracteritzen per estar suportades un total de 5 anys. S'ha de destacar que l'actualització dels packages del sistema operatiu és gratuïta, i que l'empresa destinatària del projecte en té un ampli coneixement i experiència. El paradigma LAMP s'hi pot establir sense cap tipus d'incidència.
- *Red Hat Enterprise Linux Server*: un SO igual d'estable que l'anterior, si bé requereix registrar-lo per poder-lo actualitzar posteriorment. El preu<sup>4</sup> de la subscripció més econòmica són 264€ anuals, i la desitjable 604€ anuals.

### 3.5. Anàlisi de costos/beneficis i riscos de les diferents solucions

#### Anàlisi de costos/beneficis del sistema

- Ubuntu Server LTS amb paradigma LAMP: la pila Apache HTTPd + MySQL + PHP surt a cost 0. La preparació de l'entorn existent s'ha calculat en 5 hores, a 50€/hora del tècnic. El cost de manteniment del servidor ja està pressupostat, amb el que tampoc suposa una despesa addicional per l'empresa.

Total: 250€

- Red Hat Enterprise Linux Server amb paradigma LAMP: per la modalitat *Self-Support Subscription* l'import són de 264€ anuals, i per la *Standard Subscription* 604€ anuals. El paradigma Apache HTTPd + MySQL + PHP surt a cost 0. El cost de manteniment del servidor ja està pressupostat, amb el que tampoc suposa una despesa addicional per l'empresa.

Total: entre 704€ i 1104€, segons la modalitat de subscripció

En totes dues opcions, els beneficis són idèntics: disposar d'un sistema operatiu estable, basat en GNU/Linux, i perfectament integrable en el paradigma LAMP. L'actualització del sistema (patches, updates, upgrades) és molt més àgil que en altres solucions propietàries (per exemple, entorns Windows).

---

3 Acrònim de *Long Time Support*, <https://wiki.ubuntu.com/LTS>

4 Informació extreta de <https://www.redhat.com/apps/store/server/>

### Riscos en les alternatives del sistema

- En ambdós casos, a nivell de sistema operatiu es podria requerir suport puntual en determinats escenaris; tant Ubuntu Server<sup>5</sup> com Red Hat<sup>6</sup> ofereixen diversos tipus de subscripcions de suport pels seus productes.
- Pel que fa als productes que sostenen la resta de la pila LAMP, es podria donar el cas teòric que les comunitats/empreses que hi ha darrera discontinuïn el producte. En cas que això succeís, per exemple en la política que Oracle pot prendre amb MySQL, es buscarien alternatives com poden ser migrar la capa de persistència a MariaDB<sup>7</sup> o PostgreSQL<sup>8</sup>. Qualsevol canvi suposaria assumir un cost econòmic, amb el que amb l'ús d'una solució àmpliament estesa com LAMP ens assegurem (fins on és possible) la continuïtat de l'entorn.

### 3.6. Selecció de la solució

Per tots els motius exposats anteriorment, la solució escollida és el paradigma LAMP amb Ubuntu Server LTS com a sistema operatiu de l'entorn.

---

5 Professional support from Canonical, the Ubuntu experts, <http://www.ubuntu.com/business/server/overview>. Els preus anuals són de \$320 de pel suport *Essential*, \$700 per *Standard* i \$1200 per *Advanced*.

6 Red Hat Support, <https://www.redhat.com/apps/store/server/>. Els preus anuals són de \$349 per la modalitat *Self-Support Subscription*, \$799 per *Standard Subscription* i \$1299 per *Premium Subscription*.

7 Informació del projecte MariaDB disponible a <http://mariadb.org/>. Es pot trobar un anàlisi de compatibilitat entre MySQL i MariaDB al recurs <http://kb.askmonty.org/en/mariadb-versus-mysql>.

8 Informació del projecte PostgreSQL disponible a <http://www.postgresql.org/>.

## 4. Anàlisi

### 4.1. Definició del sistema

A continuació es descriurà amb detall els requisits exactes del sistema web, introduïts en la fase anterior de viabilitat.

- La informació s'haurà de presentar de manera senzilla i a la vegada atractiva. La navegació per l'aplicació web haurà de ser instintiva, i fàcil d'executar la operativa implementada.
- Els estils quedaran definits als fulls d'estil CSS, els quals s'utilitzaran intensivament a l'aplicació web.
- Es definiran dos rols principals, administrador i usuari. Únicament els administradors dels servidors podran accedir a totes les funcionalitats de l'aplicació. Els usuaris per la seva part només podran consultar els accessos que té disponibles, i fer peticions per nous accessos, les quals requeriran del vist i plau dels administradors. És prioritària la implementació de la operativa del rol administrador per sobre de la del rol d'usuari, ja que en cas de no disposar de suficient temps, la gestió de peticions pot seguir realitzant-se des de l'actual eina de Tickets de l'empresa.
- Només podran accedir a l'aplicació persones autoritzades. L'autorització es realitzarà mitjançant usuari i contrasenya al propi sistema. La validació amb OpenLDAP és secundària, i s'intentarà implementar si es disposa de temps.
- El contingut del lloc web quedarà ordenat en quatre grans apartats, corresponents als subsistemes descrits a l'anterior punt 1.3. La operativa que es pugui executar en cada apartat quedarà definida dins dels subapartats corresponents. En cas de necessitat, l'aplicació haurà de poder acceptar nous subapartats.
- L'aplicació ha de permetre realitzar l'operativa bàsica<sup>9</sup> d'identificació, de gestió d'usuaris, de gestió de servidors i d'accessos. A més, tant els usuaris com els servidors s'han de poder agrupar en conjunts d'usuaris i servidors.
- Els accessos s'han de poder realitzar entre un usuari i un servidor, entre un usuari i un conjunt de servidors, entre un conjunt d'usuaris i un servidor, i entre un conjunt d'usuaris i un conjunt de servidors.
- L'aplicació web ha de ser compatible especialment amb el navegador Mozilla Firefox, ja que és que que l'empresa utilitza per defecte. També es valorarà la compatibilitat amb Internet Explorer i Chrome/Chromium, si bé no és prioritari.
- Les comunicacions entre l'aplicació i els servidors als quals habilitarà accés per SSH mitjançant PKI, es realitzaran a través de SSH, amb el que el servidor que conté l'aplicació haurà de tenir visibilitat de tota la resta de servidors pel port 22/tcp. La configuració de la capa de comunicació SSH amb la resta de servidors queda fora de l'abast del projecte.
- L'accés al servidor que conté l'aplicació estarà controlat per firewall a nivell de la xarxa de comunicacions, per tal de securitzar-ne les connexions entrants. No hi haurà accés des de fora de l'empresa (des d'Internet). La configuració del firewall queda fora de l'abast d'aquest projecte.

---

9 Alta, baixa i modificació.



- La llicència de l'aplicació web serà GPL<sup>10</sup> v3.

Pel que fa a l'entorn tecnològic de l'aplicació web, tindrem:

- Sistema operatiu GNU/Linux, distribució Ubuntu Server LTS.
- L'aplicació web estarà fonamentada en el paradigma LAMP, ja que està molt extès, és conegut per l'empresa i és una solució madura.
- El desenvolupament pel que fa a la programació de l'aplicació es realitzarà amb HTML+PHP.

Les normes i estàndards a seguir en la implementació del sistema web seran les següents:

- La preparació del servidor es realitzarà en base a la documentació interna de l'empresa, seguint tots els estàndards que s'aplica a qualsevol dels servidors que gestiona.
- Donada la idiosincràsia del projecte, la documentació generada es basarà en el document de la UOC Projecte web, corresponent a l'àmbit de Programari lliure, i realitzat per Alberto Otero García.

El personal involucrat en la definició de requisits i l'acceptació final, es redueix a:

- Departament d'Infraestructura, responsables del servei que ofereix la solució actual de gestió d'accessos, i a més gestors de tota la xarxa de l'empresa i els elements que conté (firewalls, VLANs, VPNs, etc).

## 4.2. Establiment de requisits

Com ja s'ha comentat a l'anterior punt 1.2, pàgines 4 i 5, el sistema web de l'aplicació queda dividit en quatre grans *subsistemes*: *identificació*, *gestió d'usuaris*, *gestió de servidors* i *gestió d'accessos*. A part de seguir una divisió a nivell lògic, l'ús de subsistemes permet facilitar-ne la gestió, la comprensió i el treball independent, així com reduir la complexitat de les tasques a realitzar .

- Anàlisi funcional del *subsistema d'identificació* :

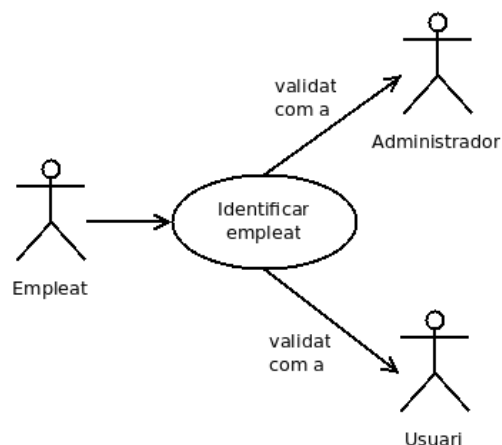


Fig. 3 : cas d'ús del subsistema d'identificació.

<sup>10</sup> Informació sobre la llicència GPL v3 disponible a <http://www.gnu.org/licenses/quick-guide-gplv3.ca.html>

<b>Cas d'ús</b>	<b>Identificar empleat</b>
<b>Actors</b>	Empleat.
<b>Funcionalitat</b>	Validar l'empleat.
<b>Resum</b>	Permetre a l'empleat accedir a l'aplicació si hi està autoritzat. En funció del rol que l'empleat tingui assignat (pot ser administrador o bé usuari), s'habilitarà l'accés a un conjunt de funcionalitats o un altre.
<b>Precondició</b>	L'usuari ha d'estar donat d'alta a l'aplicació, ja que altrament no serà possible accedir-hi.
<b>Postcondició</b>	L'actor està validat, i pot accedir a les funcionalitats que té assignades segons el seu rol.

- Anàlisi funcional del *subsistema de gestió d'usuaris* :

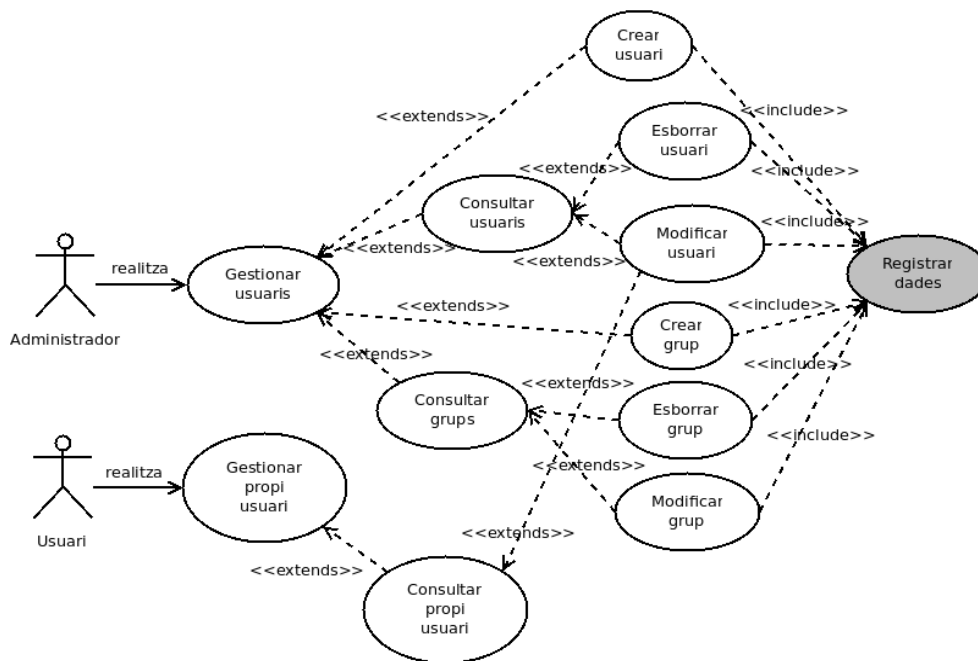


Fig. 4 : casos d'ús del subsistema de gestió d'usuaris.

<b>Cas d'ús</b>	<b>Gestionar usuaris</b>
<b>Actors</b>	Administrador.
<b>Funcionalitat</b>	Agrupar totes les funcionalitats per tipus de rol.
<b>Resum</b>	Sota aquest cas d'ús es mostraran totes les funcionalitats de gestió d'usuaris pròpies del rol administrador: crear un nou usuari, consultar qualsevol usuari, crear un nou grup d'usuaris, i consultar qualsevol grup d'usuaris. Les funcionalitats de consulta d'usuaris i grups, posteriorment permetran modificar usuaris/grups i/o eliminar-los.
<b>Precondició</b>	L'empleat ha de tenir rol d'administrador i ha d'estar validat a l'aplicació.
<b>Postcondició</b>	L'administrador té disponible l'execució de qualsevol de les funcionalitats de gestió d'usuaris definides pel seu rol.

<b>Cas d'ús</b>	<b>Crear usuari</b>
<b>Actors</b>	Administrador.
<b>Funcionalitat</b>	Crear un nou usuari.
<b>Resum</b>	Crear un nou usuari dins de l'aplicació. En el procés de creació, a part de dades personals, hi constarà la contrasenya d'accés i el tipus de rol assignat.
<b>Precondició</b>	L'usuari que es va a crear no existeix i l'administrador està validat a l'aplicació.
<b>Postcondició</b>	S'ha creat l'usuari desitjat.

<b>Cas d'ús</b>	<b>Consultar usuaris</b>
<b>Actors</b>	Administrador.
<b>Funcionalitat</b>	Consultar usuaris.
<b>Resum</b>	Consultar el detall de dades dels usuaris donats d'alta a l'aplicació. En retornarà un o més d'un, en funció del filtratge aplicat a la cerca. A partir del resultat obtingut d'usuaris, es podran modificar o bé esborrar: veure casos d'ús Esborrar usuari i modificar usuari.
<b>Precondició</b>	L'administrador està validat a l'aplicació.
<b>Postcondició</b>	Es retorna una llista amb els usuaris que coincideixen amb el criteri de cerca.

<b>Cas d'ús</b>	<b>Esborrar usuari</b>
<b>Actors</b>	Administrador.
<b>Funcionalitat</b>	Esborrar un usuari.
<b>Resum</b>	Esborrar un usuari existent dins de l'aplicació.
<b>Precondició</b>	L'usuari que es vol esborrar ha d'existir prèviament, i l'administrador ha d'estar validat a l'aplicació.
<b>Postcondició</b>	S'ha eliminat l'usuari desitjat de l'aplicació.

<b>Cas d'ús</b>	<b>Modificar usuari</b>
<b>Actors</b>	Administrador i usuari.
<b>Funcionalitat</b>	Modificar un usuari.
<b>Resum</b>	Modificar un usuari existent dins de l'aplicació.
<b>Precondició</b>	L'usuari que es vol modificar ha d'existir prèviament, i l'administrador ha d'estar validat a l'aplicació.
<b>Postcondició</b>	S'ha modificat l'usuari desitjat de l'aplicació.

<b>Cas d'ús</b>	<b>Crear grup</b>
<b>Actors</b>	Administrador.
<b>Funcionalitat</b>	Crear un nou grup d'usuaris.
<b>Resum</b>	Crear un nou grup d'usuaris dins de l'aplicació. En el procés de creació, a part dels usuaris a assignar, hi constarà el rol que podran prendre dins l'aplicació: usuaris donats d'alta a un grup amb rol d'administració podran gestionar els accessos del/s servidor/s administrat/s.
<b>Precondició</b>	El grup que es va a crear no existeix i l'administrador està validat a l'aplicació.
<b>Postcondició</b>	S'ha creat el grup d'usuaris desitjat.

<b>Cas d'ús</b>	<b>Consultar grups</b>
<b>Actors</b>	Administrador.
<b>Funcionalitat</b>	Consultar grups d'usuaris.
<b>Resum</b>	Consultar el detall d'un grup d'usuaris i qui el conforma, ja que en retorna també la llista d'usuaris. A partir del resultat obtingut de grups, es podran modificar o bé esborrar: veure casos d'ús esborrar grup i modificar grup.
<b>Precondició</b>	L'administrador està validat a l'aplicació.
<b>Postcondició</b>	Es retorna les dades del grup d'usuaris en cas d'haver-lo trobat, així com el llistat d'usuaris que en forma part.

<b>Cas d'ús</b>	<b>Esborrar grup</b>
<b>Actors</b>	Administrador.
<b>Funcionalitat</b>	Esborrar un grup d'usuaris.
<b>Resum</b>	Esborrar un grup d'usuaris existent dins de l'aplicació.
<b>Precondició</b>	El grup d'usuaris a esborrar ha d'existir prèviament, i l'administrador ha d'estar validat a l'aplicació.
<b>Postcondició</b>	S'ha eliminat el grup d'usuaris desitjat de l'aplicació.

<b>Cas d'ús</b>	<b>Modificar grup</b>
<b>Actors</b>	Administrador.
<b>Funcionalitat</b>	Modificar un grup d'usuaris.
<b>Resum</b>	Modificar un grup d'usuaris existents dins de l'aplicació.
<b>Precondició</b>	El grup d'usuaris que es vol modificar ha d'existir prèviament, i l'administrador ha d'estar validat a l'aplicació.
<b>Postcondició</b>	S'ha modificat el grup d'usuaris desitjat de l'aplicació.

<b>Cas d'ús</b>	<b>Gestionar propi usuari</b>
<b>Actors</b>	Usuari.
<b>Funcionalitat</b>	Agrupar totes les funcionalitats per tipus de rol.
<b>Resum</b>	Sota aquest cas d'ús es mostraran totes les funcionalitats de gestió d'usuaris pròpies del rol usuari: es redueix a consultar les seves pròpies dades, i poder modificar certs atributs com pot ser la contrasenya d'accés a l'aplicació.
<b>Precondició</b>	L'empleat ha de tenir rol d'usuari i ha d'estar validat a l'aplicació.
<b>Postcondició</b>	L'usuari té disponible l'execució de certes funcionalitats de gestió del seu propi usuari.

<b>Cas d'ús</b>	<b>Consultar propi usuari</b>
<b>Actors</b>	Usuari.
<b>Funcionalitat</b>	Consultar usuari.
<b>Resum</b>	Consultar el detall de dades del propi usuari que està accedint a l'aplicació.
<b>Precondició</b>	L'usuari està validat a l'aplicació.
<b>Postcondició</b>	Es retorna les dades de l'usuari que està validat a l'aplicació.

<b>Cas d'ús</b>	<b>Registrar dades</b>
<b>Actors</b>	Administrador i usuari
<b>Funcionalitat</b>	Persistir les dades de l'aplicació.
<b>Resum</b>	Consolidar qualsevol canvi que es faci amb les dades que conté l'aplicació. El registre es farà tant a nivell de DB, com d'accions a realitzar en servidors remots en cas de necessitat.
<b>Precondició</b>	L'empleat ha d'estar validat a l'aplicació.
<b>Postcondició</b>	S'ha consolidat el canvi realitzat a les dades de l'aplicació.

- Anàlisi funcional del *subsistema de gestió de servidors* :

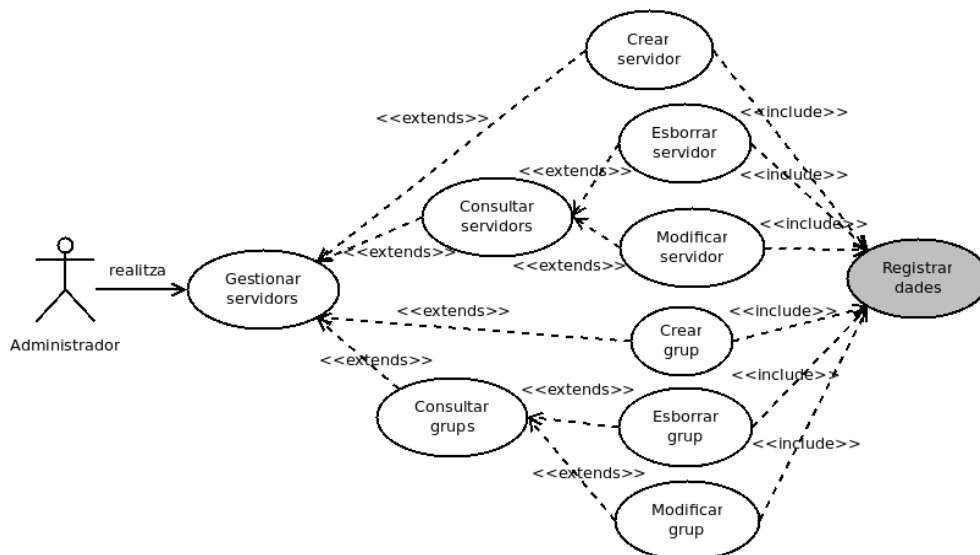


Fig. 5 : casos d'ús del subsistema de gestió de servidors.

<b>Cas d'ús</b>	<b>Gestionar servidors</b>
<b>Actors</b>	Administrador.
<b>Funcionalitat</b>	Agrupar totes les funcionalitats de gestió de servidors.
<b>Resum</b>	Sota aquest cas d'ús es mostraran totes les funcionalitats de gestió de servidors úniques del rol administrador: crear un nou servidor, consultar qualsevol servidor, crear un nou grup de servidors, i consultar qualsevol grup de servidors. Les funcionalitats de consulta de servidors i grups, posteriorment permetran modificar servidors/grups i/o eliminar-los.
<b>Precondició</b>	L'empleat ha de tenir rol d'administrador i ha d'estar validat a l'aplicació.
<b>Postcondició</b>	L'administrador té disponible l'execució de qualsevol de les funcionalitats de gestió d'usuaris definides pel seu rol.

<b>Cas d'ús</b>	<b>Crear servidor</b>
<b>Actors</b>	Administrador.
<b>Funcionalitat</b>	Crear un nou servidor.
<b>Resum</b>	Crear un nou servidor dins de l'aplicació. En el procés de creació, a part de dades del servidor, hi constarà el grup dels empleats que en són gestors.
<b>Precondició</b>	El servidor que es va a crear no existeix i l'administrador està validat a l'aplicació.
<b>Postcondició</b>	S'ha creat el servidor desitjat.

<b>Cas d'ús</b>	<b>Consultar servidors</b>
<b>Actors</b>	Administrador.
<b>Funcionalitat</b>	Consultar servidors.
<b>Resum</b>	Consultar el detall de dades dels servidors donats d'alta a l'aplicació. En retornarà un o més d'un, en funció del filtratge aplicat a la cerca. A partir del resultat obtingut de servidors, es podran modificar o bé esborrar: veure casos d'ús esborrar usuari i modificar usuari.
<b>Precondició</b>	L'administrador està validat a l'aplicació.
<b>Postcondició</b>	Es retorna una llista amb els servidors que coincideixen amb el criteri de cerca.

<b>Cas d'ús</b>	<b>Esborrar servidor</b>
<b>Actors</b>	Administrador.
<b>Funcionalitat</b>	Esborrar un servidor.
<b>Resum</b>	Esborrar un servidor existent dins de l'aplicació.
<b>Precondició</b>	El servidor que es vol esborrar ha d'existir prèviament, i l'administrador ha d'estar validat a l'aplicació.
<b>Postcondició</b>	S'ha eliminat el servidor desitjat de l'aplicació.

<b>Cas d'ús</b>	<b>Modificar servidor</b>
<i>Actors</i>	Administrador.
<i>Funcionalitat</i>	Modificar un servidor.
<i>Resum</i>	Modificar un servidor existent dins de l'aplicació.
<i>Precondició</i>	El servidor que es vol modificar ha d'existir prèviament, i l'administrador ha d'estar validat a l'aplicació.
<i>Postcondició</i>	S'ha modificat el servidor desitjat de l'aplicació.

<b>Cas d'ús</b>	<b>Crear grup</b>
<i>Actors</i>	Administrador.
<i>Funcionalitat</i>	Crear un nou grup de servidors.
<i>Resum</i>	Crear un nou grup de servidors dins de l'aplicació.
<i>Precondició</i>	El grup que es va a crear no existeix i l'administrador està validat a l'aplicació.
<i>Postcondició</i>	S'ha creat el grup de servidors desitjat.

<b>Cas d'ús</b>	<b>Consultar grups</b>
<i>Actors</i>	Administrador.
<i>Funcionalitat</i>	Consultar grups de servidors.
<i>Resum</i>	Consultar el detall d'un grup de servidors i qui el conforma, ja que en retorna també la llista de servidors. A partir del resultat obtingut de grups, es podran modificar o bé esborrar: veure casos d'ús esborrar grup i modificar grup.
<i>Precondició</i>	L'administrador està validat a l'aplicació.
<i>Postcondició</i>	Es retorna les dades del grup de servidors en cas d'haver-lo trobat, així com el llistat de servidors que en forma part.

<b>Cas d'ús</b>	<b>Esborrar grup</b>
<i>Actors</i>	Administrador.
<i>Funcionalitat</i>	Esborrar un grup de servidors.
<i>Resum</i>	Esborrar un grup de servidors existent dins de l'aplicació.
<i>Precondició</i>	El grup de servidors a esborrar ha d'existir prèviament, i l'administrador ha d'estar validat a l'aplicació.
<i>Postcondició</i>	S'ha eliminat el grup de servidors desitjat de l'aplicació.

<b>Cas d'ús</b>	<b>Modificar grup</b>
<b>Actors</b>	Administrador.
<b>Funcionalitat</b>	Modificar un grup de servidors.
<b>Resum</b>	Modificar un grup de servidors existents dins de l'aplicació.
<b>Precondició</b>	El grup de servidors que es vol modificar ha d'existir prèviament, i l'administrador ha d'estar validat a l'aplicació.
<b>Postcondició</b>	S'ha modificar el grup de servidors desitjat de l'aplicació.

<b>Cas d'ús</b>	<b>Gestionar propi usuari</b>
<b>Actors</b>	Usuari.
<b>Funcionalitat</b>	Agrupar totes les funcionalitats per tipus de rol.
<b>Resum</b>	Sota aquest cas d'ús es mostraran totes les funcionalitats de gestió d'usuaris pròpies del rol usuari: es redueix a consultar les seves pròpies dades, i poder modificar certs atributs com pot ser la contrasenya d'accés a l'aplicació.
<b>Precondició</b>	L'empleat ha de tenir rol d'usuari.
<b>Postcondició</b>	L'usuari té disponible l'execució de certes funcionalitats de gestió del seu propi usuari.

<b>Cas d'ús</b>	<b>Consultar propi usuari</b>
<b>Actors</b>	Usuari.
<b>Funcionalitat</b>	Consultar usuari.
<b>Resum</b>	Consultar el detall de dades del propi usuari que està accedint a l'aplicació.
<b>Precondició</b>	L'usuari està validat a l'aplicació.
<b>Postcondició</b>	Es retorna les dades de l'usuari que està validat a l'aplicació.

<b>Cas d'ús</b>	<b>Registrar dades</b>
<b>Actors</b>	Administrador.
<b>Funcionalitat</b>	Persistir les dades de l'aplicació.
<b>Resum</b>	Consolidar qualsevol canvi que es faci amb les dades que conté l'aplicació. El registre es farà tant a nivell de DB, com d'accions a realitzar en servidors remots en cas de necessitat.
<b>Precondició</b>	L'empleat ha d'estar validat a l'aplicació.
<b>Postcondició</b>	S'ha consolidat el canvi realitzat a les dades de l'aplicació.



- Anàlisi funcional del subsistema de gestió d'accessos :

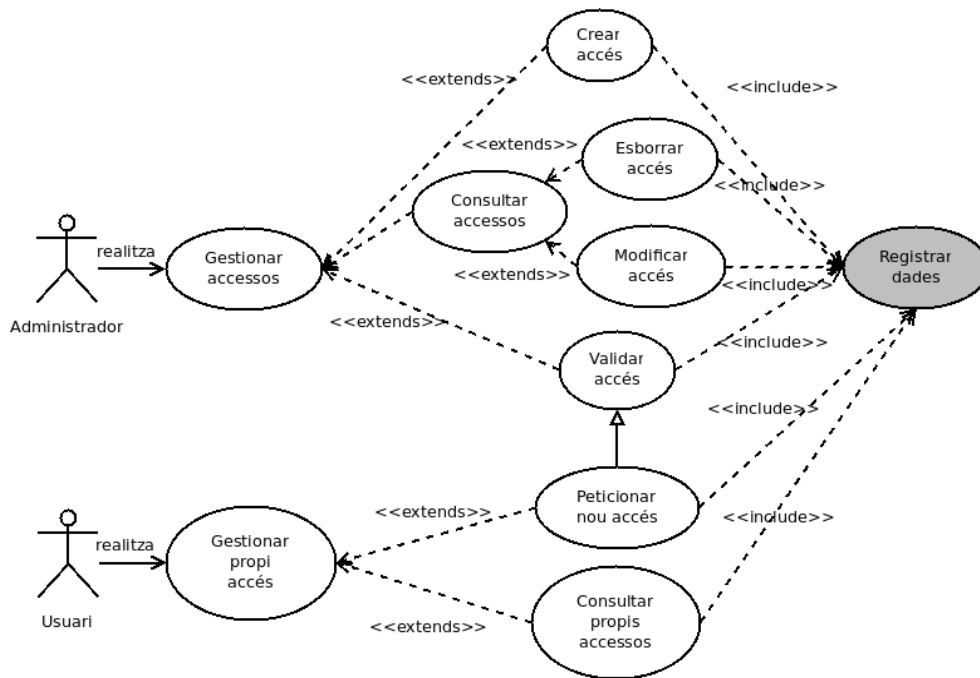


Fig. 6 : casos d'ús del subsistema de gestió d'accessos.

<b>Cas d'ús</b>	<b>Gestionar accessos</b>
<b>Actors</b>	Administrador.
<b>Funcionalitat</b>	Agrupar totes les funcionalitats de gestió d'accessos segons el rol de l'actor
<b>Resum</b>	Sota aquest cas d'ús es mostraran totes les funcionalitats de gestió d'accessos pròpies del rol administrador: crear un nou accés, consultar qualsevol accés i validar peticions d'accés realitzades per usuaris. La funcionalitat de consulta d'accessos permetrà posteriorment modificar i/o eliminar un accés.
<b>Precondició</b>	L'empleat ha de tenir rol d'administrador i ha d'estar validat a l'aplicació.
<b>Postcondició</b>	L'administrador té disponible l'execució de qualsevol de les funcionalitats de gestió d'accessos definides pel seu rol.

<b>Cas d'ús</b>	<b>Crear accés</b>
<b>Actors</b>	Administrador.
<b>Funcionalitat</b>	Crear un nou accés.
<b>Resum</b>	Crear un nou accés dins de l'aplicació. En el procés de creació, es relacionarà algun element de la tupla usuari / grup d'usuaris amb la tupla servidor / grup de servidors.
<b>Precondició</b>	L'accés que es va a crear no existeix i l'administrador està validat a l'aplicació.
<b>Postcondició</b>	S'ha creat l'accés desitjat.

<b>Cas d'ús</b>	<b>Consultar accessos</b>
<b>Actors</b>	Administrador.
<b>Funcionalitat</b>	Consultar accessos.
<b>Resum</b>	Consultar el detall dels accessos actius a l'aplicació. En retornarà un o més d'un, en funció del filtratge aplicat a la cerca. A partir del resultat obtingut d'accessos, es podran modificar o bé esborrar: veure casos d'ús esborrar accés i modificar accés.
<b>Precondició</b>	L'administrador està validat a l'aplicació.
<b>Postcondició</b>	Es retorna una llista amb els accessos que coincideixen amb el criteri de cerca.

<b>Cas d'ús</b>	<b>Esborrar accés</b>
<b>Actors</b>	Administrador.
<b>Funcionalitat</b>	Esborrar un accés.
<b>Resum</b>	Esborrar un accés existent dins de l'aplicació.
<b>Precondició</b>	L'accés que es vol esborrar ha d'existir prèviament, i l'administrador ha d'estar validat a l'aplicació.
<b>Postcondició</b>	S'ha eliminat l'accés desitjat de l'aplicació.

<b>Cas d'ús</b>	<b>Modificar accés</b>
<b>Actors</b>	Administrador.
<b>Funcionalitat</b>	Modificar un accés.
<b>Resum</b>	Modificar un accés existent dins de l'aplicació.
<b>Precondició</b>	L'accés que es vol modificar ha d'existir prèviament, i l'administrador ha d'estar validat a l'aplicació.
<b>Postcondició</b>	S'ha modificat l'accés desitjat de l'aplicació.

<b>Cas d'ús</b>	<b>Validar accés</b>
<b>Actors</b>	Administrador.
<b>Funcionalitat</b>	Validar un accés pendent.
<b>Resum</b>	Validar un accés demanat per un usuari, i que encara està pendent.
<b>Precondició</b>	Existeix una petició d'accés d'un usuari pendent de confirmar, i l'administrador ha d'estar validat a l'aplicació.
<b>Postcondició</b>	S'ha confirmat l'accés i aquest queda activat.

<b>Cas d'ús</b>	<b>Gestionar propi accés</b>
<i>Actors</i>	Usuari.
<i>Funcionalitat</i>	Agrupar totes les funcionalitats de gestió d'accessos segons el rol de l'actor.
<i>Resum</i>	Sota aquest cas d'ús es mostraran totes les funcionalitats de gestió d'accessos pròpies del rol usuari: consultar els propis accessos i peticionar un nou accés, pendent que sigui validat per un administrador.
<i>Precondició</i>	L'empleat ha de tenir rol d'usuari i ha d'estar validat a l'aplicació.
<i>Postcondició</i>	L'usuari té disponible l'execució de qualsevol de les funcionalitats de gestió d'accessos definides pel seu rol.

<b>Cas d'ús</b>	<b>Peticionar nou accés</b>
<i>Actors</i>	Usuari.
<i>Funcionalitat</i>	Peticionar un nou accés.
<i>Resum</i>	Demandar un nou accés a un servidor o conjunt de servidors. Fins que un administrador no corrobore la petició, l'usuari no disposarà de l'accés actiu.
<i>Precondició</i>	L'empleat ha de tenir rol d'usuari i ha d'estar validat a l'aplicació.
<i>Postcondició</i>	L'usuari té pendent de validar una petició d'accés.

<b>Cas d'ús</b>	<b>Registrar dades</b>
<i>Actors</i>	Administrador i usuari
<i>Funcionalitat</i>	Persistir les dades de l'aplicació.
<i>Resum</i>	Consolidar qualsevol canvi que es faci amb les dades que conté l'aplicació. El registre es farà tant a nivell de DB, com d'accions a realitzar en servidors remots en cas de necessitat.
<i>Precondició</i>	L'empleat ha d'estar validat a l'aplicació.
<i>Postcondició</i>	S'ha consolidat el canvi realitzat a les dades de l'aplicació.

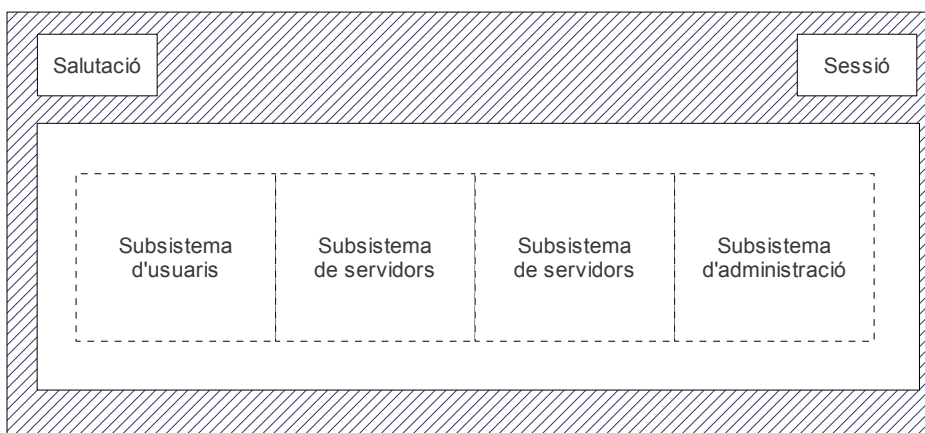
### 4.3. Identificació d'interfícies d'usuari

L'aplicació de gestió d'accessos serà utilitzada per usuaris relacionats amb els departaments d'administració de sistemes i de projectes (desenvolupadors); així, el perfil de tots ells serà netament tècnic.

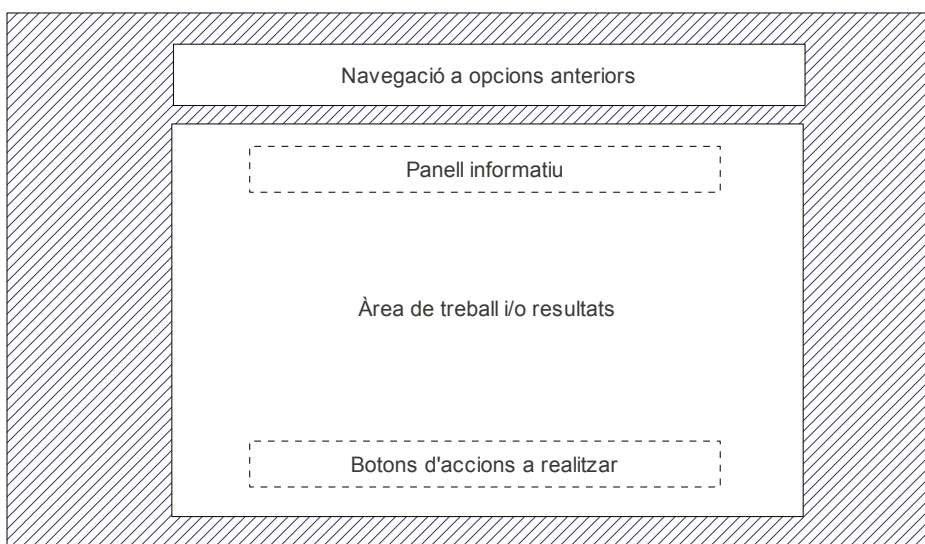
El sistema web descrit fins ara tindrà les següents característiques:

- L'accés es realitzarà des d'un navegador web, principalment Firefox.
- L'aplicació haurà de ser clara i senzilla d'utilitzar; es valorarà positivament que no s'hagin de realitzar transicions entre múltiples pantalles de forma habitual.
- Els errors detectats hauran d'estar identificats de forma clara.

La distribució de la interfície d'usuari de l'aplicació seguirà els següents diagrames:



*Fig. 7 : interfície corresponent al menú inicial de l'aplicació.*



*Fig. 8 : interfície comuna a tots els casos d'ús.*

Per tal de facilitar al màxim la interacció amb la interfície, aquest mateix plantejament serà l'utilitzat en tots els casos d'ús detectats a l'aplicació.

#### 4.4. Pla de proves

De cara a preparar el pla de proves, s'ha de tenir en compte que queda fora de l'abast del present projecte la implementació del *subsistema de comunicacions*; aquest fet implica que si bé l'aplicació acabarà comunicant-se amb servidors remots, no ho farà encara en aquesta fase inicial del projecte. Així l'aplicació únicament actua sobre la capa de persistència de dades.

Es realitzaran les següents proves a mesura que evolucioni la implementació l'aplicació web:

- Proves unitàries.
  - Abast: casos d'ús.
  - Descripció: a mesura que cada cas d'ús dels definits a l'apartat 2.2 es vagi implementant, s'hi realitzaran proves unitàries per tal de validar-ne la seva correctesa. És vital poder assegurar que el que es va implementant és vàlid abans de continuar amb els següents casos d'ús.
- Proves d'integració.
  - Abast: subsistemes.
  - Descripció: es realitzaran a nivell de subsistema, una vegada aquest disposi de totes les funcionalitats implementades. Igual que en el punt anterior, és essencial que es validi la fiabilitat d'un subsistema abans que aquest comenci a interactuar amb la resta de subsistemes.
- Proves de sistema.
  - Abast: sistema web complet
  - Descripció: validar que totes aquelles operatives que requereixen d'interacció entre diferents subsistemes, també s'executen de forma correcta. Serà la darrera validació funcional de l'aplicació, dins de l'àmbit de la persistència de dades, i excloent el de comunicació amb la resta de servidors.
- Proves d'implantació: crash recovery de la DB.
  - Abast: base de dades MySQL
  - Descripció: validar que el sistema de backup que s'hagi implementat a nivell de persistència de dades MySQL realment funciona, tot fent un *crash-recovery* de la DB. Aquesta validació s'haurà de documentar per presentar a l'empresa. En cas que no es disposi de temps suficient, la prova de crash recovery la realitzarà la mateixa empresa.

## 5. Disseny

### 5.1. Definició de l'arquitectura

Mitjançant la següent notació UML es defineix l'arquitectura de la solució proposada, formada per tres grans sistemes:

- *Sistema gestor d'accessos*: corresponent a l'aplicació que s'implementa en el present projecte.
- *Sistema de persistència*: referent a la persistència de dades requerida pel sistema gestor d'accessos.
- *Sistema de servidors*: engloba el conjunt de servidors l'accés del qual és gestionat pel sistema d'aplicació.

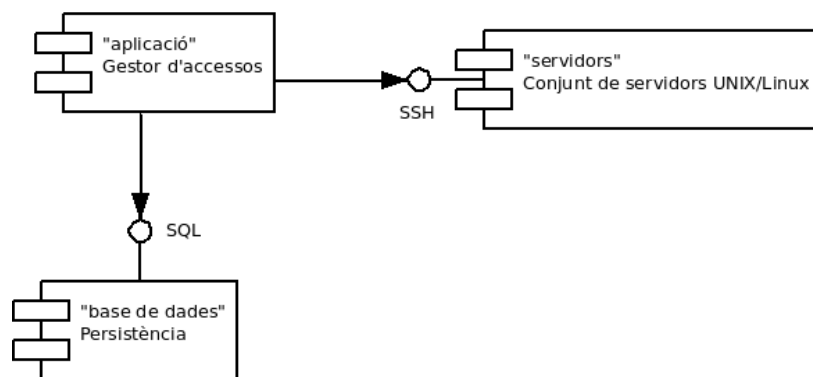


Fig. 9 : comunicacions establertes entre sistemes

### 5.2. Concreció dels subsistemes

En apartats anteriors s'han anat concretant els subsistemes d'*usuaris*, de *servidors*, d'*accessos* i d'*identificació*, com a parts del sistema global d'aplicació. A més, existeixen tres agrupacions addicionals:

- *Sistema de persistència de dades*: a diferència dels quatre anteriors, no es desenvoluparà a mida, ja que s'utilitzarà una solució de SGBD com és MySQL.
- *Subsistema de comunicació*: format per un conjunt d'scripts GNU/Linux que s'ocupen de permetre la comunicació entre l'aplicació i la resta de servidors als quals copia les claus PKI per posteriorment permetre'n l'accés sense contrasenya. La programació d'aquest conjunt d'scripts, adequació dels existents a la nova aplicació i/o desenvolupament des de zero amb PHP surt de l'àmbit del projecte.
- *Subsistema d'administració*: en funció de les necessitats dels scripts del subsistema de *comunicació*, serà precís desenvolupar a mida un subsistema que s'ocupi de gestionar la configuració interna de l'aplicació.

Els subsistemes queden interrelacionats de la següent forma:

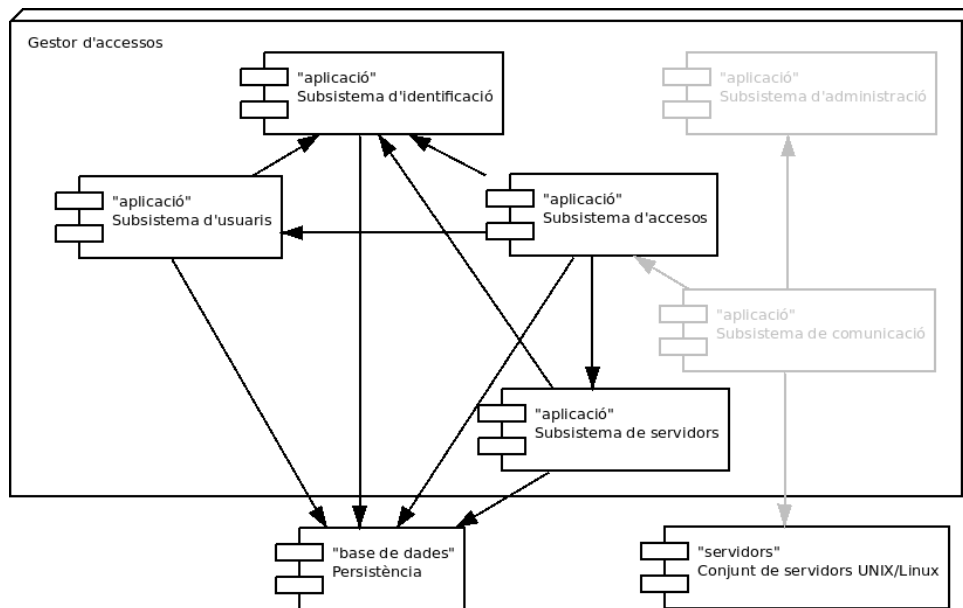


Fig. 10 : comunicacions establertes entre subsistemes

A partir del diagrama anterior, s'observa que:

- Els *subsistemes d'usuaris*, *servidors* i *accessos* requereixen del *subsistema d'identificació*, ja que no permeten executar cap acció si prèviament no existeix una sessió vàlida.
- El *subsistema d'identificació* requereix de la capa de persistència, fet que implica que la tupla usuari / contrasenya d'accés a l'aplicació es guardarà a la base de dades. En cas d'afegir una nova funcionalitat com pot ser la validació mitjançant OpenLDAP, la dependència actual queda anul·lada, establint-ne en el seu lloc una de nova contra el subsistema extern de gestió de *directori*.
- La persistència de dades dels subsistemes *d'usuaris*, *servidors* i *accessos* es realitza dins del subsistema d'estereotip "base de dades".
- La gestió d'*accessos* requereix de comunicació amb els *subsistemes d'usuaris* i de *servidors*, per tal de poder definir els propis accessos.
- Els *subsistemes de comunicació* i *d'administració* queden fora de l'àmbit inicial del projecte: es desenvoluparan posteriorment.

En quant als tipus de comunicació:

- La comunicació que s'estableix amb la capa de persistència és de tipus SQL.
- Les dades del *subsistema d'identificació* requerides per la resta de subsistemes es transferiran mitjançant l'ús de `SESSION`.
- Tant les comunicacions internes com les establertes entre els *subsistemes d'usuaris, servidors i accessos* utilitzen el mètode de tipus `POST`.

### 5.3. Arquitectura del sistema d'aplicació

S'ha escollit el patró de Model Vista Controlador (MVC) per definir l'arquitectura interna del sistema d'aplicació. El patró MVC separa l'aplicació en tres capes:

- *Model*: engloba el conjunt de dades amb el qual opera l'aplicació, permetent tant a la vista com al controlador mostrar i tractar tota a informació gestionada.
- *Vista*: ofereix una presentació coherent i adequada del model de dades a l'usuari mitjançant un conjunt d'interfícies gràfiques. És la capa amb la qual l'usuari de l'aplicació interactua amb la mateixa.
- *Controlador*: conté tota la lògica de negoci, reaccionant d'una determinada forma a les accions iniciades per part de l'usuari. La capa de controladors s'ocupa d'accedir a la informació del model de dades, i facilitar-les a les vistes per tal que siguin mostrades.

L'objectiu que busca l'estratificació és facilitar el control de l'aplicació, el manteniment i la implementació de noves millores amb un impacte mínim sobre la resta de capes. A més, en projectes més grans, on hi ha un equip de persones treballant, el model MVC permet crear aplicacions de forma més ràpida i efectiva que en mètodes tradicionals, ja que diferents grups de desenvolupadors poden estar treballant en paral·lel en les tres capes del model, sense afectar-se mútuament.

S'han analitzat a nivell teòric varis *framework* MVC per PHP, i finalment s'ha escollit TinyMVC<sup>11</sup>, el qual dona resposta a dues necessitats del projecte: aprofitar els avantatges que ofereix el patró MVC i obtenir una aplicació senzilla de seguir i de mantenir a nivell de codificació (veure punt 2.3).

TinyMVC es caracteritza per separar el model de la vista, i comunicar-les únicament mitjançant el controlador. És una diferència arquitectònica respecte altres *framework*, on sí es permet un accés directe des de la vista al model, però per contra deixes de tenir centralitzada tota la lògica de negoci al controlador.

---

11 <http://www.tinymvc.com>



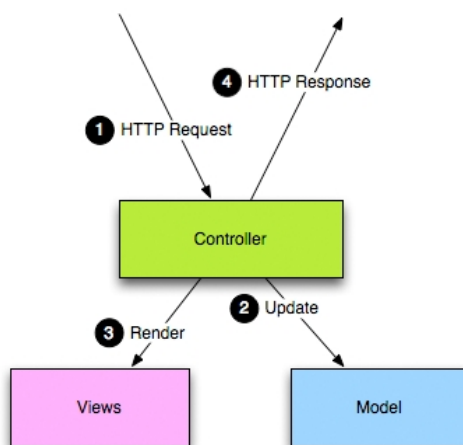


Fig. 11 : exemple d'operació d'actualització executada pel model MVC de TinyMVC

Amb TinyMVC es busca simplicitat i obtenció ràpida de resultats, invertint el menor temps possible en la corba d'aprenentatge implícita a tot *framework*, per tal que l'empresa pugui mantenir fàcilment l'aplicació amb uns costos el més reduïts possibles.

#### 5.4. Diagrames de casos d'ús

Una vegada començada la fase de disseny, s'acorda amb l'empresa no implementar l'entorn de l'actor *Usuari* (veure punt 3.2), ja que és un requeriment de baixa prioritat. Així, cap dels casos d'ús del rol *Usuari* s'implementaran.

També es realitza la següent modificació al *subsistema d'identificació*, creant un nou cas d'ús de canvi de contrasenya:

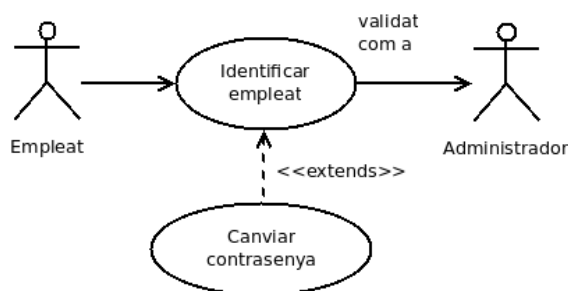


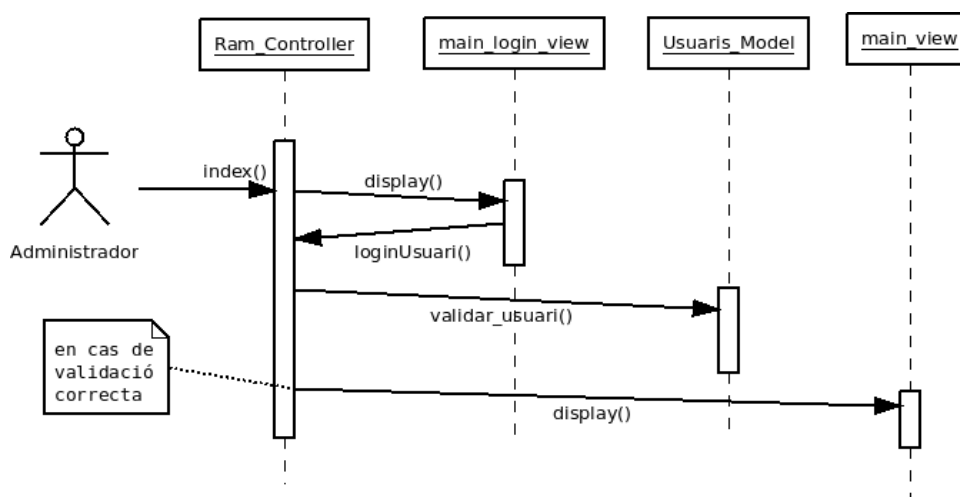
Fig. 12 : cas d'ús del subsistema d'identificació.

<b>Cas d'ús</b>	<b>Canviar contrasenya</b>
<b>Actors</b>	Empleat.
<b>Funcionalitat</b>	Canviar la contrasenya de l'empleat.
<b>Resum</b>	Permetre a l'empleat poder canviar la seva contrasenya, sempre i quan faciliti correctament les credencials d'accés a l'aplicació
<b>Precondició</b>	L'empleat ha d'estar donat d'alta a l'aplicació, ja que altrament no serà possible validar-se i canviar la contrasenya.
<b>Postcondició</b>	L'actor ha canviat la contrasenya d'accés a l'aplicació, però no ha entrat a la mateixa.

La resta de casos d'ús exposats al punt 3.2 es mantenen.

## 5.5. Diagrames addicionals

A continuació s'exposen diagrames de seqüència pels casos d'ús més representatius del *sistema d'aplicació*, fonamentats en el patró MVC que implementa el *framework* TinyMVC:



**Fig. 13** : diagrama de seqüència del cas d'ús identificar empleat, amb validació correcta.

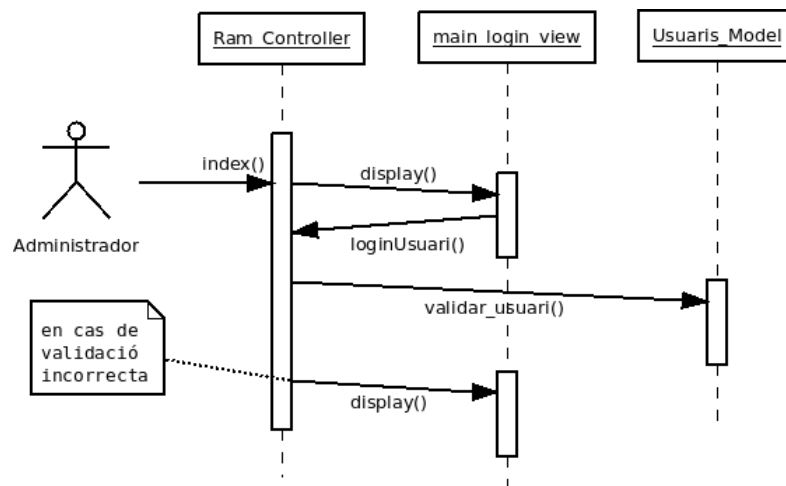


Fig. 14 : diagrama de seqüència del cas d'ús identificar empleat, amb validació incorrecta.

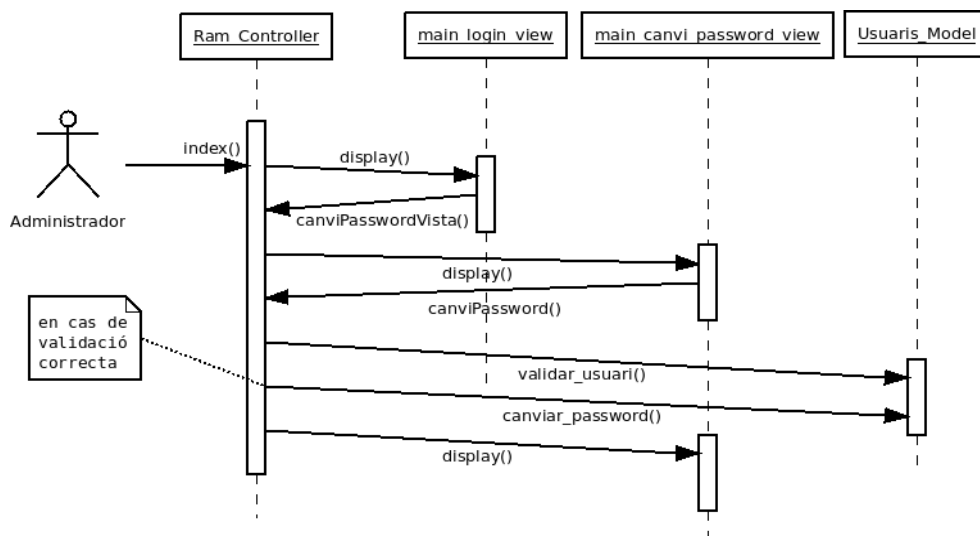


Fig. 15 : diagrama de seqüència del cas d'ús canvi contrasenya, amb validació correcta.

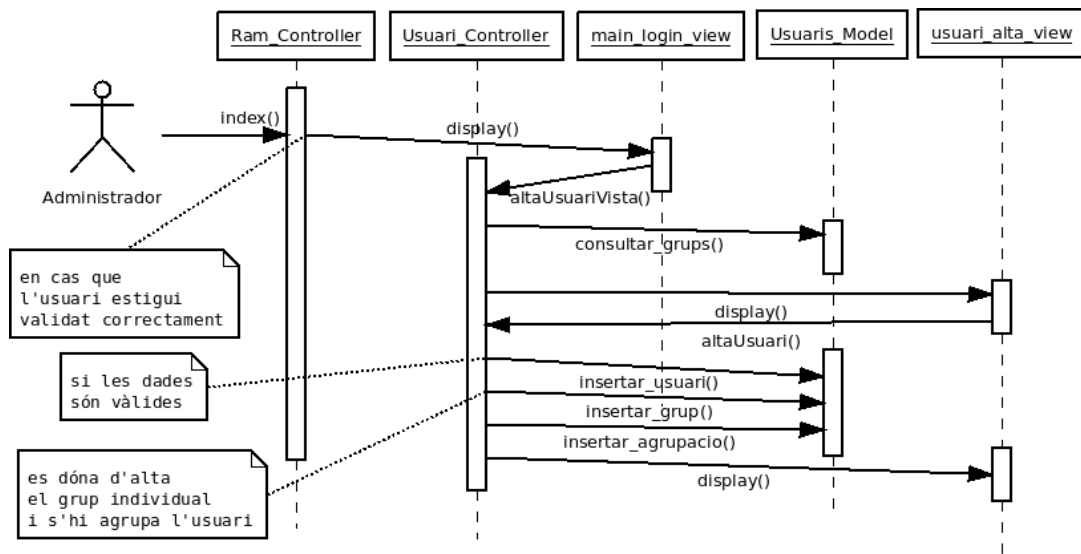


Fig. 16 : diagrama de seqüència del cas d'ús crear usuari.

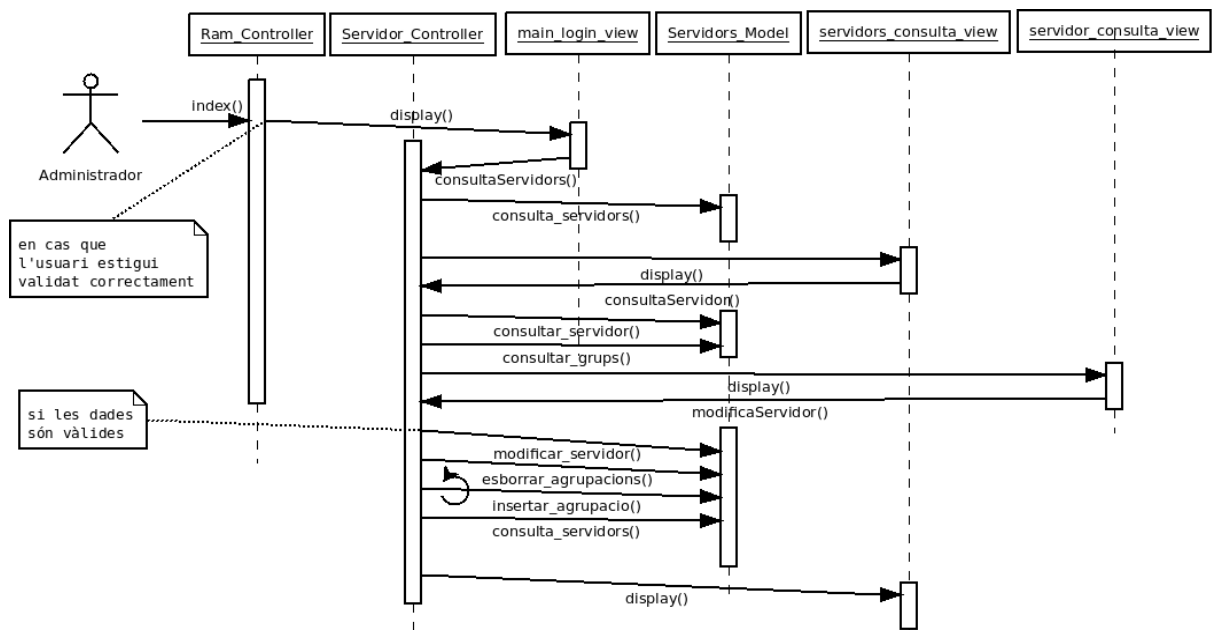


Fig. 17 : diagrama de seqüència del cas d'ús modificar servidor.

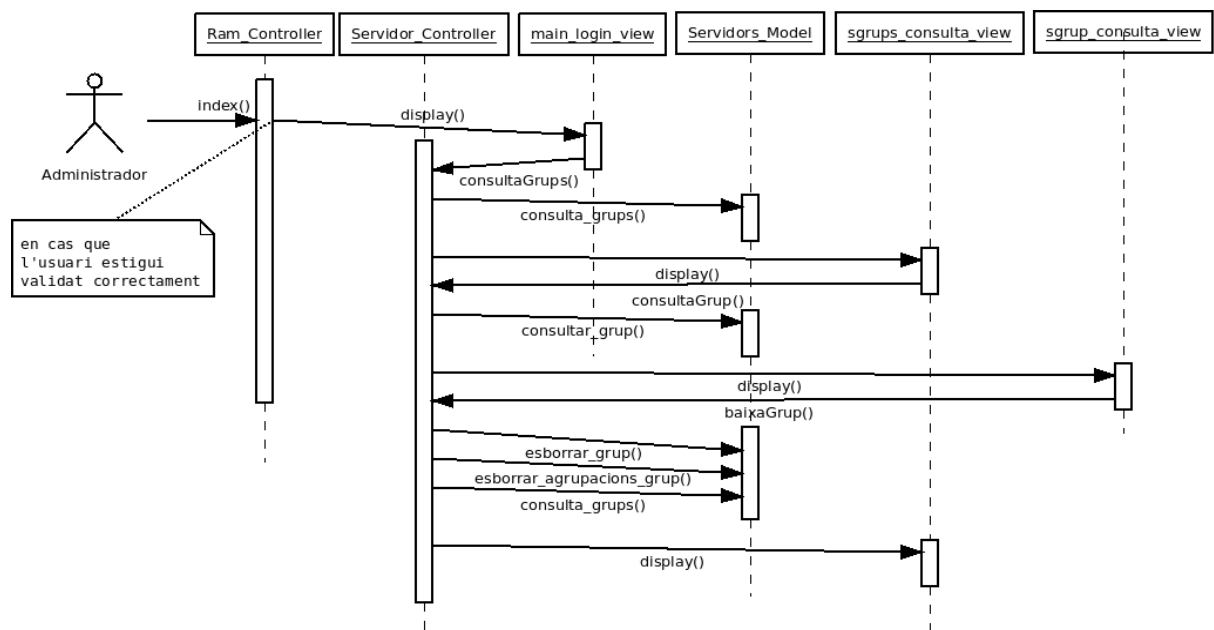


Fig. 18 : diagrama de seqüència del cas d'ús esborrar grup de servidors.

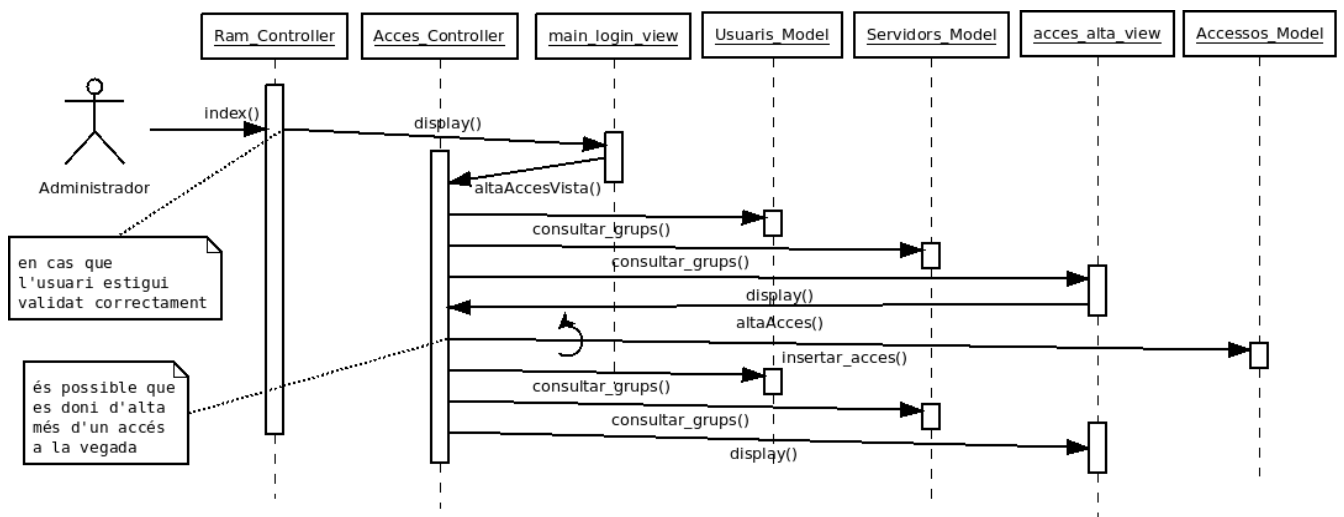


Fig. 19 : diagrama de seqüència del cas d'ús crear accés.

### 5.6. Diagrama de classes

Les entitats que conformen l'aplicació són les següents:

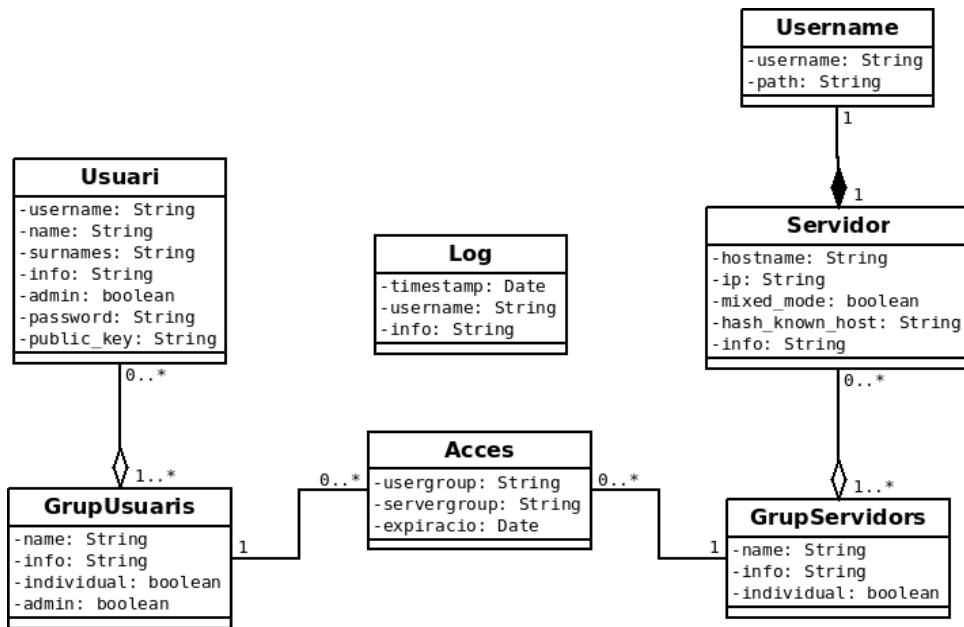


Fig. 20 : diagrama de classes, escenari general.

### 5.7. Diagrama ER

En base a l'esquema de l'apartat anterior, s'arriba al següent diagrama ER:

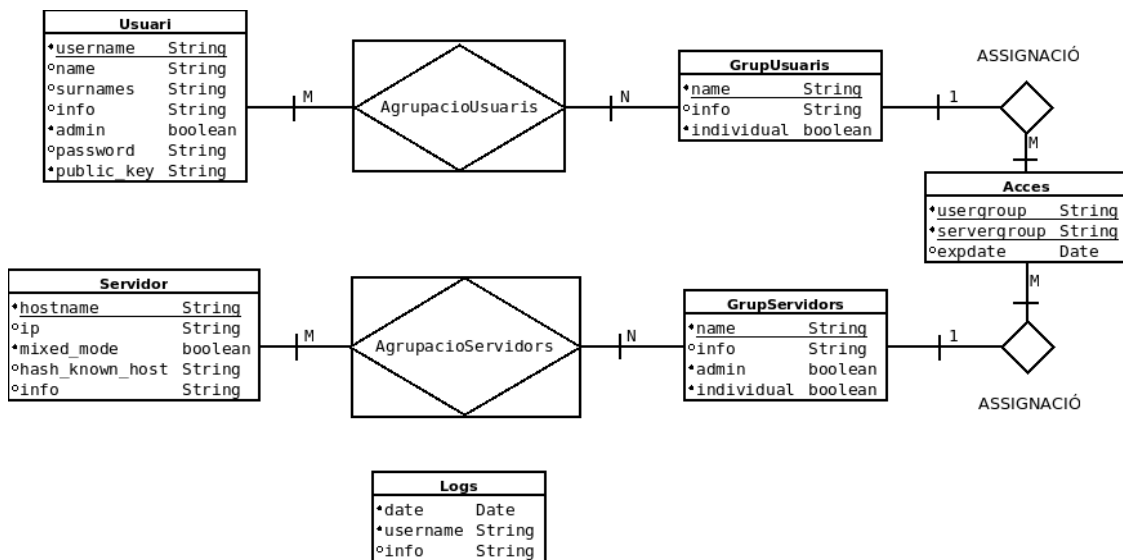


Fig. 21 : diagrama ER, escenari general.

## 5.8. Disseny de la capa de persistència

Taula Usuaris:

<i>Camp</i>	<i>Tipus</i>	<i>Clau</i>	<i>Descripció</i>
<i>username</i>	CHAR(40)	PK	Username, identificador d'usuari.
<i>name</i>	CHAR(30)		Nom.
<i>surnames</i>	CHAR(60)		Cognoms.
<i>info</i>	TEXT		Descripció.
<i>admin</i>	TINYINT(1)		Indica si és o no administrador.
<i>password</i>	CHAR(32)		Contrasenya.
<i>public_key</i>	TEXT		Clau pública PKI.

Taula Servidors:

<i>Camp</i>	<i>Tipus</i>	<i>Clau</i>	<i>Descripció</i>
<i>hostname</i>	CHAR(40)	PK	Hostname, identificador de servidor.
<i>ip</i>	CHAR(16)		Adreça IP.
<i>mixed_mode</i>	TINYINT(1)		Mode de distribució de PKI als servidors destí: si és 1 (true), permet un doble ús de l'arxiu <code>authorized_keys</code> .
<i>hash_known_host</i>	CHAR(32)		Signatura del servidor destí.
<i>user</i>	CHAR(10)		Usuari amb el qual es connecta al servidor destí.
<i>path</i>	CHAR(30)		Path de l'usuari destí on s'ubica l'arxiu <code>authorized_keys</code> ( <code>~/.ssh/authorized_keys</code> ).
<i>info</i>	TEXT		Descripció.

Taula GrupUsuaris:

<i>Camp</i>	<i>Tipus</i>	<i>Clau</i>	<i>Descripció</i>
<i>name</i>	CHAR(20)	PK	Nom, identificador del grup d'usuaris.
<i>info</i>	TEXT		Descripció.
<i>admin</i>	TINYINT(1)		Indica si és o no un grup administrador.
<i>individual</i>	TINYINT(1)		Indica si és o no un grup individual: un grup d'un únic usuari, el nom del qual és idèntic al de l'usuari.

Taula GrupServidors:

<i>Camp</i>	<i>Tipus</i>	<i>Clau</i>	<i>Descripció</i>
<i>name</i>	CHAR(20)	PK	Nom, identificador del grup de servidors.
<i>info</i>	TEXT		Descripció.
<i>individual</i>	TINYINT(1)		Indica si és o no un grup individual: un grup d'un únic servidor, el nom del qual és idèntic al del servidor.

Taula AgrupacioUsuaris:

Camp	Tipus	Clau	Descripció
hostname	CHAR(40)	PK, FK	Identificador de l'usuari.
groupname	CHAR(20)	PK, FK	Identificador del grup d'usuaris.

Taula AgrupacioServidors:

Camp	Tipus	Clau	Descripció
servername	CHAR(40)	PK, FK	Identificador del servidor.
groupname	CHAR(20)	PK, FK	Identificador del grup d'usuaris.

Taula Accessos:

Camp	Tipus	Clau	Descripció
usergroup	CHAR(20)	PK, FK	Identificador del grup d'usuaris.
servergroup	CHAR(20)	PK, FK	Identificador del grup de servidors
dataexp	TINYINT(1)		Data d'expiració del permís.

Taula Logs:

Camp	Tipus	Clau	Descripció
date	TIMESTAMP		Timestamp de les accions registrades al log.
username	CHAR(40)		Usuari que ha realitzat l'acció registrada.
info	TEXT		Detall de l'acció realitzada.

## 5.9. Creació de la interfície gràfica

Les vistes creades són:

Vista	Descripció
<code>acces_alta_view.php</code>	Vista on es dona d'alta un accés a l'aplicació. El formulari que conté apunta al mètode <code>altaAcces()</code> del controlador <code>Acces_Controller</code> . Disposa d'un enllaç al menú inicial de l'aplicació.
<code>acces_consulta_view.php</code>	Vista on es mostra el detall de tots els atributs d'un accés determinat. El formulari que conté té un doble apuntador als mètodes <code>modificaAcces()</code> i <code>baixaAcces()</code> del controlador <code>Acces_Controller</code> . Disposa d'enllaços directes d'accés al menú inicial de l'aplicació i a la pantalla general de consulta d'accessos.



<i>Vista</i>	<i>Descripció</i>
<i>accessos_consulta_view.php</i>	Vista on es mostren els accessos donats d'alta a l'aplicació. El formulari apunta al mètode consultaAcces() del controlador Acces_Controller. Disposa d'un enllaç directe al menú inicial.
<i>main_canvi_password_view.php</i>	Vista on es permet canviar la contrasenya d'un administrador. El formulari apunta al mètode canviPassword() del controlador Ram_Controller. Disposa d'un enllaç directe a la pantalla inicial d'identificació.
<i>main_login_view.php</i>	Vista on es mostra la pantalla inicial d'identificació de l'usuari a l'aplicació. El formulari que conté té un doble apuntador als mètodes loginUsuari() i canviPasswordVista() del controlador Ram_Controller.
<i>main_view.php</i>	Vista que conté el menú principal de l'aplicació, des d'on es pot donar d'alta/consultar un servidor, un grup de servidors, un usuari, un grup d'usuaris o un accés, utilitzant el controlador corresponent. Les opcions de modificació i eliminació d'un determinat element sempre es gestionaran des de les opcions prèvies de consulta. Disposa d'un enllaç de desconnexió de l'aplicació (tancament de la sessió) que retorna a la vista main_login_view.php.
<i>servidor_alta_view.php</i>	Vista on es dona d'alta un servidor a l'aplicació. El formulari que conté apunta al mètode altaServidor() del controlador Acces_Controller. Disposa d'un enllaç al menú inicial de l'aplicació.
<i>servidor_consulta_view.php</i>	Vista on es mostra el detall de tots els atributs d'un servidor concret. El formulari que conté té un doble apuntador als mètodes modificaServidor() i baixaServidor() del controlador Servidor_Controller. Disposa d'enllaços directes d'accés al menú inicial de l'aplicació i a la pantalla general de consulta de servidors.
<i>servidors_consulta_view.php</i>	Vista on es mostren els servidors donats d'alta a l'aplicació. El formulari apunta al mètode consultaServidor() del controlador Servidor_Controller. Disposa d'un enllaç directe al menú inicial.
<i>sgrup_alta_view.php</i>	Vista on es dona d'alta un grup de servidors a l'aplicació. El formulari que conté apunta al mètode altaGrup() del controlador Servidor_Controller. Disposa d'un enllaç al menú inicial de l'aplicació.
<i>sgrup_consulta_view.php</i>	Vista on es mostren els atributs d'un grup de servidors determinat. El formulari que conté té un doble apuntador als mètodes modificaGrup() i baixaGrup() del controlador Servidor_Controller. Disposa d'enllaços directes d'accés al menú inicial de l'aplicació i a la pantalla general de consulta de grups de servidors.
<i>sgrups_consulta_view.php</i>	Vista on es mostren els grups de servidors donats d'alta a l'aplicació. El formulari apunta al mètode consultaGrup() del controlador Servidor_Controller. Disposa d'un enllaç directe al menú inicial.
<i>ugrup_alta_view.php</i>	Vista on es dona d'alta un grup d'usuaris a l'aplicació. El formulari que conté apunta al mètode altaGrup() del controlador Usuari_Controller. Disposa d'un enllaç al menú inicial de l'aplicació.

Vista	Descripció
<i>ugrup_consulta_view.php</i>	Vista on es mostren els atributs d'un grup d'usuaris concret. El formulari que conté té un doble apuntador als mètodes <code>modificaGrup()</code> i <code>baixaGrup()</code> del controlador <code>Usuari_Controller</code> . Disposa d'enllaços directes d'accés al menú inicial de l'aplicació i a la pantalla general de consulta de grups d'usuaris.
<i>ugrups_consulta_view.php</i>	Vista on es mostren els grups d'usuaris donats d'alta a l'aplicació. El formulari apunta al mètode <code>consultaGrup()</code> del controlador <code>Usuari_Controller</code> . Disposa d'un enllaç directe al menú inicial.
<i>usuari_alta_view.php</i>	Vista on es dona d'alta un usuari a l'aplicació. El formulari que conté apunta al mètode <code>altaUsuari()</code> del controlador <code>Usuari_Controller</code> . Disposa d'un enllaç al menú inicial de l'aplicació.
<i>usuari_consulta_view.php</i>	Vista on es mostra el detall de tots els atributs d'un usuari determinat. El formulari que conté té un doble apuntador als mètodes <code>modificaUsuari()</code> i <code>baixaUsuari()</code> del controlador <code>Usuari_Controller</code> . Disposa d'enllaços directes d'accés al menú inicial de l'aplicació i a la pantalla general de consulta d'usuaris.
<i>usuaris_consulta_view.php</i>	Vista on es mostren els usuaris donats d'alta a l'aplicació. El formulari apunta al mètode <code>consultaUsuari()</code> del controlador <code>Usuari_Controller</code> . Disposa d'un enllaç directe al menú inicial.

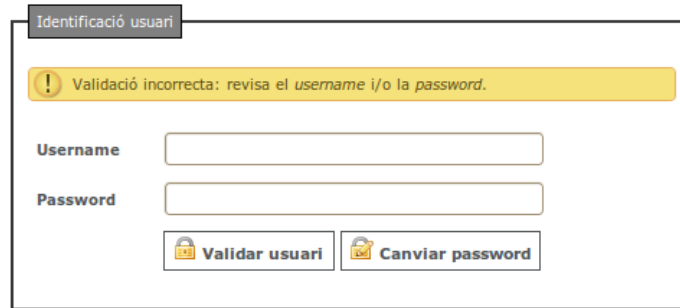
El principal objectiu que la vista en el paradigma MVC és la de mostrar a l'usuari les dades d'aplicació gestionada pel model de dades, i poder executar les accions que tingui associades. A més, seguint amb les indicacions de l'apartat 2.3, la interfície ha de ser senzilla d'utilitzar, intuïtiva, agradable i fàcil de mantenir, motius pels quals s'ha utilitzat un *full d'estils* CSS.

### Subsistema d'identificació

La vista d'identificació permet validar l'usuari o bé canviar-ne la contrasenya:

Fig. 22 : vista d'identificació d'usuari.

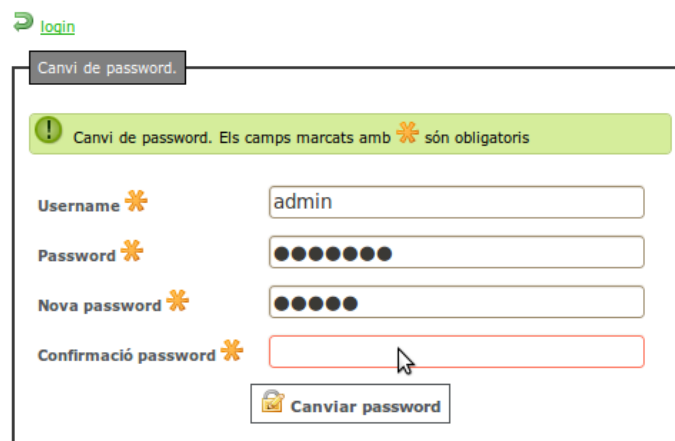
En cas que no s'hagi validat correctament, es mostra el missatge informatiu corresponent.



The screenshot shows a web form titled "Identificació usuari". At the top, there is a yellow error message box with an exclamation mark icon and the text: "Validació incorrecta: revisa el username i/o la password." Below the message are two input fields: "Username" and "Password". At the bottom of the form, there are two buttons: "Validar usuari" (with a lock icon) and "Canviar password" (with a lock icon).

Fig. 23 : vista d'identificació incorrecta d'usuari.

A la vista de canvi de contrasenya, s'han d'especificar tant la contrasenya actual, com la nova amb llur confirmació:



The screenshot shows a web form titled "Canvi de password." with a green header bar containing a warning icon and the text: "Canvi de password. Els camps marcats amb \* són obligatoris". The form has four input fields: "Username \*" (containing "admin"), "Password \*" (with masked characters), "Nova password \*" (with masked characters), and "Confirmació password \*" (empty). A "Canviar password" button with a lock icon is at the bottom.

Fig. 24 : vista de canvi de contrasenya.

Una vegada validat l'usuari com a administrador, es mostra la vista del menú principal de l'aplicació:

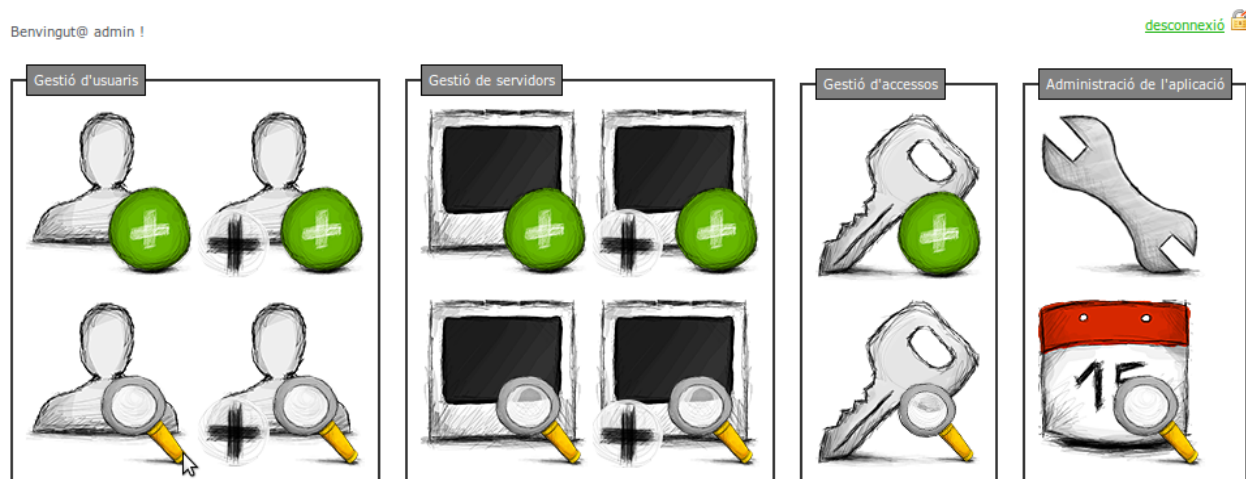
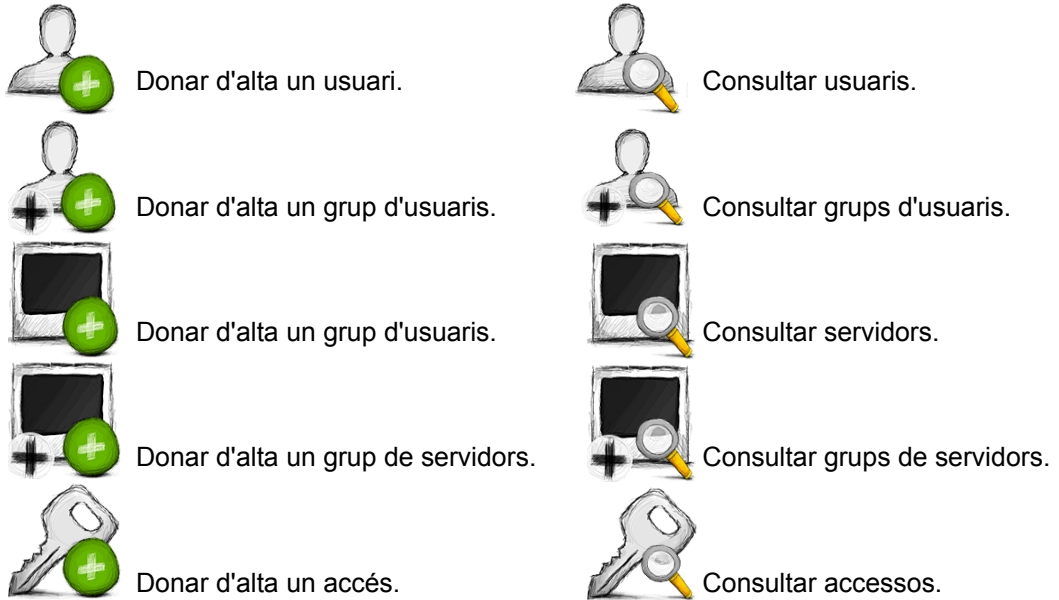


Fig. 25 : vista del menú principal.

El menú principal s'ha definit seguint la separació lògica dels subsistemes de l'aplicació. L'aplicació base del present projecte té únicament implementades les gestions d'*usuaris*, de *servidors* i d'*accessos*; la darrera opció d'administració de l'aplicació està orientada pel *subsistema d'administració*, el qual com ja s'ha indicat anteriorment no s'implementa en aquesta entrega.

El significat de cada opció implementada és:



Subsistema d'usuaris

→ menu

Alta usuari

! Donar d'alta un usuari al sistema. Els camps obligatoris estan marcats amb \*

Username \*

Nom

Cognoms

Administrador

Password \*

Confirmació password \*

Descripció

Clau pública \*

Grups  BO-INF  BO-SAC

Fig. 26 : vista d'alta d'usuari.

En funció del resultat de l'operació, s'adequa el missatge informatiu resultant:

Alta usuari

! Usuari 'jordi.blasco.planesas' donat d'alta correctament.

Alta usuari

! Els valors username i/o clau pública no poden ser buits

Fig. 27 : missatges informatius comuns a totes les vistes.

A totes les vistes on es mostra un conjunt d'elements (*usuaris, servidors, grups d'usuaris, grups de servidors, accessos*) s'ha habilitat una funcionalitat per cer cerques per un determinat camp:

menu

Consulta usuaris

! Consulta d'usuaris. Es pot filtrar la sortida amb l'opció cercar.

Cercar :  Username

Username	Nom	Cognoms	Descripció
admin			
jordi.blasco.planesas	Jordi	Blasco Planesas	

Fig. 28 : vista d'usuaris.

La vista de consulta d'usuari permet modificar valors o donar de baixa l'usuari:

menu - consulta usuaris

Modificació usuari

! Es pot *modificar* l'usuari i desar els canvis amb  (els camps \* són obligatoris) o bé *eliminar-lo* amb .

Username \*

Nom

Cognoms

Administrador

Descripció

Clau pública \*

Grups  BO-INF  BO-SAC

Fig. 29 : vista de consulta d'usuaris.

La gestió de grups d'usuaris segueix la mateixa pauta que la gestió d'usuaris:

menu

Alta grup d'usuaris

! Donar d'alta un grup d'usuaris al sistema. Els camp obligatori està marcat amb \*  
Un grup administrador permet administrar els servidors que gestiona.

Nom \* BO-INF

Descripció Infraestructura

Administrador

Afegir grup Cancelar

Fig. 30 : vista d'alta de grup d'usuaris.

menu

Consulta grups usuaris

! Grup 'BO-INF' modificat correctament.

Cercar :  Nom

Nom	Descripció	Admin
BO-INF	Infraestructura	<input type="button" value="🔍"/>
BO-SAC		<input type="button" value="🔍"/>
jordi.blasco.planesa	grup individual	

Fig. 31 : vista de grups d'usuaris.

menu - consulta grups

Modificació grup

! Es pot *modificar* el grup i desar els canvis amb   
o bé *eliminar-lo* amb .

Nom BO-SAC

Descripció

Administrador

Modificar grup Eliminar grup

Fig. 32 : vista de consulta d'usuari.

### Subsistema de servidors

El subsistema de servidors segueix la línia d'estil i funcionalitat marcada per les vistes del *subsistema d'usuaris*. A continuació s'exposen algunes de les vistes:

Fig. 33 : vista d'alta de servidor.

Hostname	IP	Descripció
haendel	188.25.32.15	servidor de correu
mozart		servidor OpenLDAP

Fig. 34 : vista de consulta de servidors.


Hostname	IP	Descripció
haendel	188.25.32.15	servidor de correu

Fig. 35 : vista de consulta de servidors filtrada.


### Subsistema d'accessos


La vista d'alta d'accés permet realitzar assignacions múltiples a la vegada: un o varis grups d'usuaris donats d'alta per un o varis grups de servidors.

Els grups queden representats de la següent manera:

 : grup individual d'usuaris.

 : grup individual de servidors.

 : grup d'usuaris.

 : grup de servidors.

Dues de les vistes del *subsistema d'accessos* són:

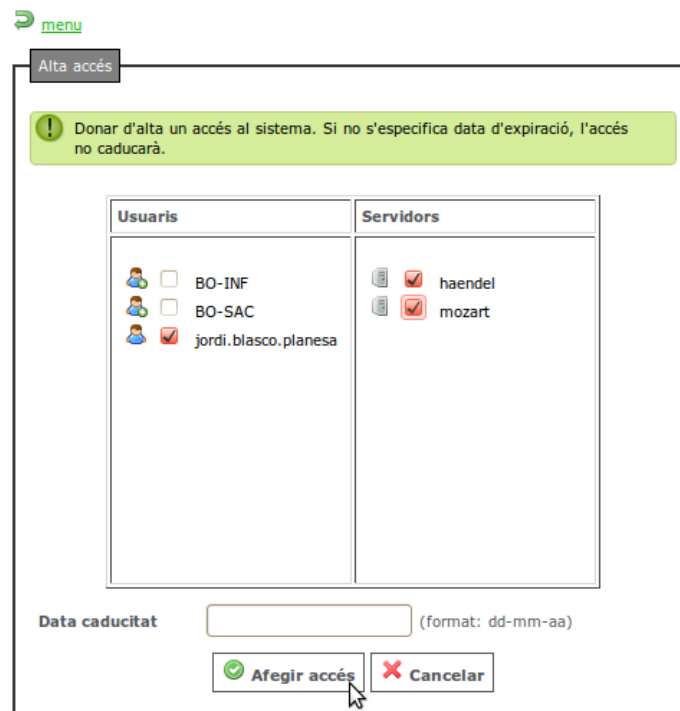


Fig. 36 : vista d'alta d'accés.

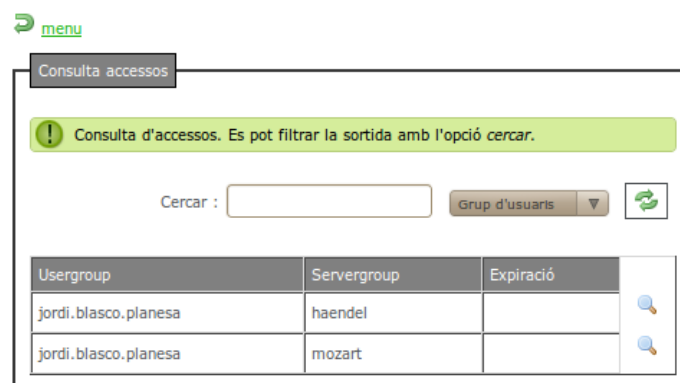


Fig. 37 : vista de consulta d'accessos.



## 6. Desenvolupament

### 6.1. Estructura

L'estructura de l'aplicació web realitzada ha seguit la disposició que per defecte duu implementada el *framework* TinyMVC :

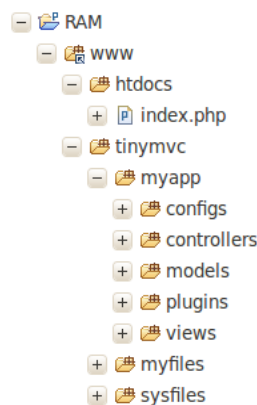


Fig. 38 : estructura de carpetes del framework TinyMVC.

L'arxiu `index.php` genèric s'utilitza per carregar tots els components inicials del *framework* i ubicar els contextos de les aplicacions. En el present cas únicament n'hi ha una, ubicada a `tinymvc/myapp`.

Un dels punts més importants a qualsevol *framework* és la possibilitat de connectar contra un SGBD. En aquesta solució, la configuració s'implementa a `tinymvc/myapp/configs/database.php` :

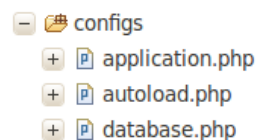


Fig. 39 : arxius de configuració del framework TinyMVC.

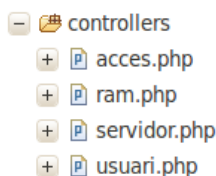
Els paràmetres a inicialitzar per poder establir connexió contra una DB són els habituals: el tipus de connexió, el nom del servidor on està desplegada l'aplicació, el nom de la base de dades, i l'usuari i contrasenya de connexió a la base de dades.

```
$config['default']['plugin'] = 'TinyMVC_PDO'; // plugin for db access
$config['default']['type'] = 'mysql'; // connection type
$config['default']['host'] = 'localhost'; // db hostname
$config['default']['name'] = 'ram'; // db name
$config['default']['user'] = 'root'; // db username
$config['default']['pass'] = 'test1'; // db password
$config['default']['persistent'] = false; // db connection persistence?
```

Fig. 40 : configuració d'accés a DB del framework TinyMVC.

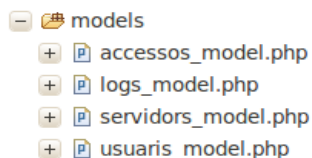
Seguint el model MVC exposat al punt 4.3, l'aplicació se separa a nivell lògic en tres capes: el model, responsable de gestionar les dades de l'aplicació, la vista, que s'ocupa de mostrar a l'usuari les dades de l'aplicació i totes aquelles opcions que pot executar, i el controlador, el *core* de l'aplicació que s'ocupa d'implementar tota la lògica del negoci.

Així, la carpeta `tinymvc/myapp/controllers` conté els controladors de l'aplicació. S'ha optat per separar en quatre controladors tota la lògica de negoci amb l'objectiu de facilitar llur implementació i, sobretot, posteriors tasques de manteniment i ampliació de l'aplicació.



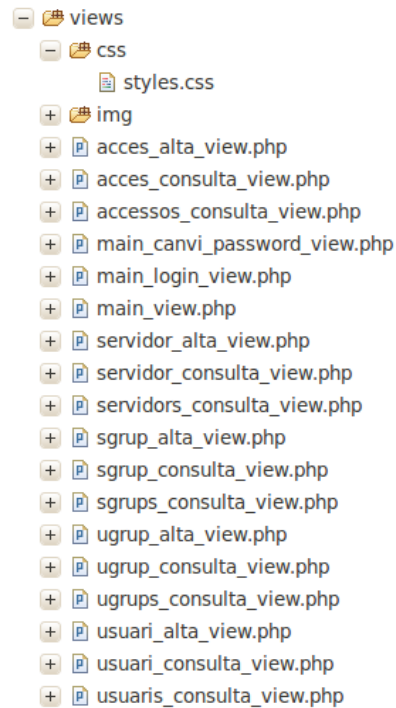
*Fig. 41 : controladors definits al framework TinyMVC.*

La gestió de les dades de l'aplicació està codificada dins de la carpeta `tinymvc/myapp/models`. Els models també estan agrupats per subsistemes amb l'excepció del *subsistema d'identificació*, que s'ha inclòs dins del *subsistema d'usuaris*. Aquesta elecció s'ha pres després de valorar el *subsistema d'identificació*: no té un conjunt de dades propi, ja que s'abasteix de les dades dels usuaris donats d'alta a l'aplicació.



*Fig. 42 : models definits al framework TinyMVC.*

La capa de vistes s'ubiquen a la carpeta `tinymvc/myapp/views`. Dins d'ella se n'hi han creat dues més, `css` i `img`, per tal de donar cabuda als *fulls d'estil* CSS de l'aplicació i al conjunt d'imatges i icones utilitzades, respectivament.



*Fig. 43 : vistes definides al framework TinyMVC.*

## 6.2. Desplegament

Donat que TinyMVC és un *framework* molt lleuger<sup>12</sup>, s'ha optat per fer l'entrega de forma conjunta del framework + aplicació de gestió d'accessos.

El seu desplegament és tan senzill com descomprimir l'arxiu `aplicacio.tar.gz` dins de la carpeta d'aplicacions definida a l'Apache HTTPd + PHP del nostre servidor, i accedir directament a la URL <http://localhost/htdocs/index.php/ram> per validar el correcte desplegament.

La pantalla inicial que es mostra és la de validació de l'aplicació:

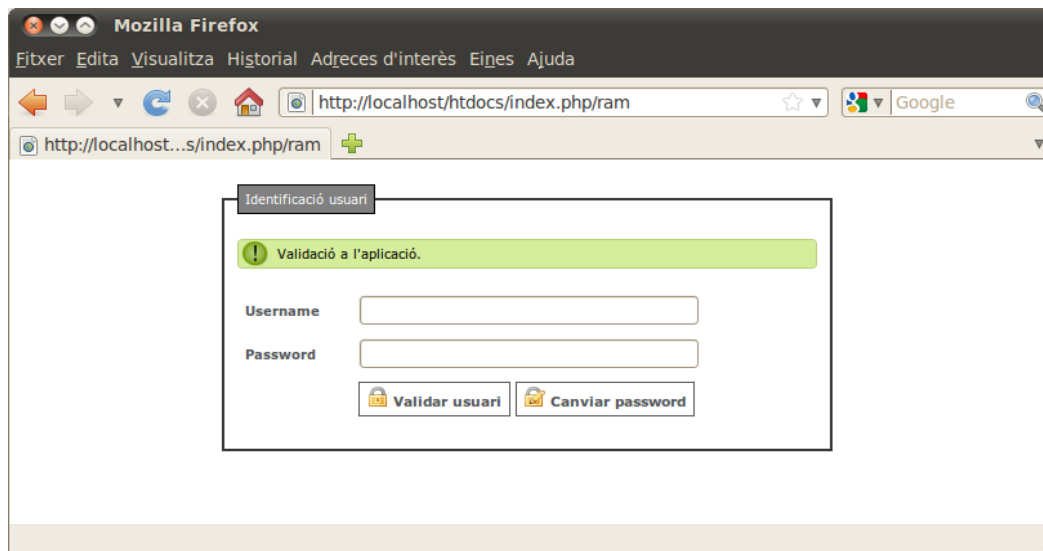


Fig. 44 : vista inicial de l'aplicació desplegada.

Per tal que sigui completament operativa, és necessari realitzar dos passos més:

- Configurar els paràmetres de connexió contra el SGBD MySQL del nostre servidor, editant l'arxiu esmentat anteriorment `tinymvc/myapp/configs/database.php`.
- Proveir d'estructura i dades la base de dades. L'script de creació de les taules MySQL requerides per l'aplicació, crea totes les estructures dins d'una `database` anomenada `ram`. Una de les possibles formes d'executar l'script de càrrega de dades és:

```
root@server:/var/www# mysql -h localhost -u root -p < ram.sql
Enter password:
root@server:/var/www#
```

L'script a més crea un usuari administrador inicial, per tal que es pugui començar a treballar amb l'aplicació. Les dades són:

```
username: admin    contrasenya13: hola123
```

<sup>12</sup> El pes de la versió TinyMVC 1.2.1 és de només 33KB comprimit, i 123KB descomprimit. Es pot descarregar directament des de <http://www.tinymvc.com/download/>.

<sup>13</sup> Per tal de millorar la seguretat d'accés a l'aplicació, totes les contrasenyes desades a la DB resten encriptades amb MD5.

## 7. Conclusions

### 7.1. Globals

El present projecte mostra com és possible implementar solucions de qualitat a necessitats reals utilitzant únicament programari lliure. Totes les fases del projecte han pogut ser implementades amb programari lliure: processador de textos, creació de diagrames UML, captura i edició d'imatges, llenguatge de programació, *framework* MVC, SGBD, servidor web, sistema operatiu, navegador web, IDE, validacions amb servidors virtuals, etc.

D'altra banda s'ha observat com el la divisió lògica del projecte en *subsistemes*, juntament amb l'ús del paradigma MVC, és realment útil a l'hora de desenvolupar ja que permet separar les funcionalitats amb capes, facilitant la codificació, reutilització i manteniment de codi. Malgrat tot, el paradigma MVC és un tipus de disseny que té més sentit a mesura que l'aplicació va creixent i es va ampliant en funcionalitats (escalabilitat), ja que en les primeres fases es requereix una gran quantitat d'arxius en relació a les funcionalitats reduïdes que s'implementen.

Durant el desenvolupament del projecte s'ha produït un contratemps important a nivell familiar. En aquesta situació de necessitat s'ha pogut comprovar com les metodologies *agile* permeten una millor gestió del temps; d'igual forma és vital poder establir una comunicació contínua amb l'empresa per tal que el client conegui el grau d'evolució de l'aplicació.

La conclusió global és que el Màster de programari lliure és útil per les necessitats del dia a dia.

### 7.2. Aplicació

En el moment de plantejar l'abast del projecte amb l'empresa, es van definir un mínim desitjable: en finalitzar el projecte s'havia de disposar d'una aplicació web bàsica que gestionés a nivell de persistència totes les dades d'usuaris, servidors i accessos. A partir d'aquest punt, en cas de disposar de més temps, s'anirien implementant els subsistemes de comunicacions i d'administració, per tal que fos plenament operativa.

La primera fase del projecte ha pogut ser implementada completament, a més d'alguna funcionalitat addicional com l'auditoria interna. Malgrat tot, l'aplicació no és completament operativa per falta del subsistema de comunicacions, que finalment ha quedat fora de l'abast del projecte per falta de temps.

El projecte es pot millorar utilitzant un *framework* MVC PHP més estès i amb més funcionalitats que no TinyMVC (ús d'AJAX, possibilitat de comunicació directa entre les vistes i el model, etc). Aquest punt, però, queda condicionat a que l'empresa accepti invertir més temps en aprendre com funciona el nou *framework* del que requereix TinyMVC.

A nivell personal el projecte m'ha aportat nous coneixements del llenguatge PHP, del qual personalment no en sabia res, així com del *framework* TinyMVC. També ha requerit d'un esforç addicional pel que fa a la gestió global del projecte, concretament amb la gestió del temps, degut a una malaltia greu familiar de llarga durada. En aquest sentit, la comunicació contínua tant amb l'empresa com amb el consultor ha estat vital; en un entorn més hostil, hagués estat complicat de superar.

Segons s'ha anat validant amb l'empresa, el projecte compleix els requeriments bàsics establerts. En base a ells es podrà completar i millorar la solució utilitzada actualment. Com que UPCnet és l'empresa on treballa actualment, una vegada feta la present entrega jo mateix podré continuar implementant i millorant l'aplicació.

## 8. Glossari

- **Subsistema** : agrupació lògica interna d'una aplicació, que conté una sèrie de components d'una mateixa naturalesa. S'utilitza per dividir l'aplicació en parts més petites, amb un grau més o menys alt d'independència, i poder així donar resposta a necessitats d'un menor grau de complexitat. El projecte actual implementa els subsistemes d'identificació, d'usuaris, de servidors i d'accessos.
- **PKI** : acrònim d'infraestructura de clau pública (Public Key Infrastructure). En el context del projecte, fa referència a criptografia de clau pública, o també anomenada criptografia asimètrica. A partir de la generació d'un parell de claus pública i privada, un usuari pot encriptar un missatge a partir de la clau pública, i posteriorment descodificar-lo utilitzant la clau privada. La relació entre claus té la característica que matemàticament la clau privada mai es pot obtenir a partir de la clau pública.
- **MVC** : patró arquitectònic que separa el model d'una aplicació (dades), la interfície amb la qual interactua l'usuari (vista), i la lògica de negoci de l'aplicació (controlador). El patró MVC es troba habitualment en aplicacions WEB, i pot ser implementat per frameworks propis de llenguatges com PHP i/o Java.
- **SGBD** : acrònim de Sistema Gestor de Base de Dades. És un tipus de programari específic que serveix d'interfície entre la base de dades, l'usuari, i les aplicacions que hi accedeixen. Entre els SGBD lliures hi destaquen MySQL, l'utilitzat en el present projecte, Postgres i MariaDB.
- **CSS** : acrònim de Cascading Steel Sheet, són fulls d'estils que tenen com a objectiu separar l'estructura d'un document de la seva presentació (forma). Els arxius CSS s'utilitzen per donar format a una pàgina web, i permet obtenir una codificació més neta i la reutilització d'estils entre els diferents elements que componen una pàgina web.
- **ER** : fa referència als models entitat-relació, diagrames que s'utilitzen pel modelat de la persistència de dades d'un sistema d'informació. En aquests models s'hi mostren les entitats rellevants i les seves interrelacions i propietats.
- **Framework** : representa una arquitectura de programari que modela les relacions generals entre totes les entitats que conformen el domini, i són dissenyats amb l'objectiu de facilitar el desenvolupament de les aplicacions.
- **Script** : programa molt senzill que es pot executar des de la línia de comandes del propi sistema operatiu. Habitualment es desen en simples arxius de text, i són interpretats línia a línia en temps d'execució.
- **MD5** : acrònim de Message-Digest Algorithm 5. Algoritme criptogràfic que permet obtenir una sortida característica (*resum*) per cada bloc de dades que encripta.
- **CPD** : acrònim de Centre de Processament de Dades, i fa referència a la ubicació on resideixen els servidors d'una organització.
- **IDE** : acrònim de Integrated Development Environment, és un programari que facilita les tasques de programació als desenvolupadors. Permet operacions com l'edició de codi font, compilació, execució, debug, etc. Pel present projecte s'ha utilitzat l'IDE Eclipse for PHP Developers.

## 9. Bibliografia

OTERO GARCIA, Alberto. Programari Lliure: Projecte Web. 1a edició, Barcelona: Fundació per a la Universitat Oberta de Catalunya, febrer 2006.

CAMPS PARÉ, Rafael, et al. Programari Lliure: Base de dades. 2a edició, Barcelona: Fundació per a la Universitat Oberta de Catalunya, febrer 2007.

MATEU, Carles. Programari Lliure: Desenvolupament d'aplicacions web. 1a edició, Barcelona: Fundació per a la Universitat Oberta de Catalunya, febrer 2006.

CAMPS PARÉ, Rafael, et al. Bases de dades I. 3a edició, Barcelona: Fundació per a la Universitat Oberta de Catalunya, setembre 2005.

DOMINGO FERER, Josep; HERRERA JOANCOMARTÍ, Jordi; RIFÀ POUS, Helena. Criptografia. 3a edició, Barcelona: Eureka Media S.L., febrer 2006

PHP (<http://www.php.net/>)

Framework TinyMVC (<http://www.tinymvc.com/>)

AgileManifesto (<http://agilemanifesto.org/>)

AgileAlliance (<http://www.agilealliance.org/>)

## 10. Annexes

### 10.1. Sistema de logs

Un dels requisits indicats a l'apartat 2.3 és el d'auditar les operacions executades per l'aplicació. Tot i ser de baixa prioritat, ha donat temps d'implementar el registre d'accions dins de la taula Logs de MySQL; exemple:

```
mysql> select * from Logs;
```

date	username	info
2012-01-20 16:09:01	admin	Alta usuari [jordi]
2012-01-20 16:09:01	admin	Alta grup usuari individual [jordi]
2012-01-20 16:09:13	admin	Alta usuari [raquel]
2012-01-20 16:09:13	admin	Alta grup usuari individual [raquel]
2012-01-20 16:09:25	admin	Alta usuari [yolanda]
2012-01-20 16:09:25	admin	Alta grup usuari individual [yolanda]
2012-01-20 16:09:40	admin	Alta usuari [paco]
2012-01-20 16:09:40	admin	Alta grup usuari individual [paco]
2012-01-20 16:09:51	admin	Alta usuari [marga]
2012-01-20 16:09:51	admin	Alta grup usuari individual [marga]
2012-01-20 16:10:12	admin	Alta grup usuari [B01]
2012-01-20 16:10:17	admin	Alta grup usuari [B02]
2012-01-20 16:10:24	admin	Modificat usuari [jordi]
2012-01-20 16:10:24	admin	Desassociat usuari [jordi] de tots els grups
2012-01-20 16:10:24	admin	Associat usuari [jordi] a grup [B01]
2012-01-20 16:10:28	admin	Modificat usuari [marga]
2012-01-20 16:10:28	admin	Desassociat usuari [marga] de tots els grups
2012-01-20 16:10:28	admin	Associat usuari [marga] a grup [B02]
2012-01-20 16:10:35	admin	Modificat usuari [paco]
2012-01-20 16:10:35	admin	Desassociat usuari [paco] de tots els grups
2012-01-20 16:10:35	admin	Associat usuari [paco] a grup [B02]
2012-01-20 16:10:39	admin	Modificat usuari [raquel]
2012-01-20 16:10:39	admin	Desassociat usuari [raquel] de tots els grups
2012-01-20 16:10:39	admin	Associat usuari [raquel] a grup [B01]
2012-01-20 16:10:43	admin	Modificat usuari [yolanda]
2012-01-20 16:10:43	admin	Desassociat usuari [yolanda] de tots els grups
2012-01-20 16:10:43	admin	Associat usuari [yolanda] a grup [B01]
2012-01-20 16:11:03	admin	Alta grup servidor [infraestructura]
2012-01-20 16:11:06	admin	Alta grup servidor [serveis]
2012-01-20 16:11:23	admin	Alta servidor [mozart]
2012-01-20 16:11:23	admin	Alta grup servidor individual [mozart]
2012-01-20 16:11:23	admin	Associat servidor [mozart] a grup [infraestructura]
2012-01-20 16:11:29	admin	Alta servidor [haendel]
2012-01-20 16:11:29	admin	Alta grup servidor individual [haendel]
2012-01-20 16:11:29	admin	Associat servidor [haendel] a grup [infraestructura]
2012-01-20 16:11:42	admin	Alta servidor [dvorak]
2012-01-20 16:11:42	admin	Alta grup servidor individual [dvorak]
2012-01-20 16:11:42	admin	Associat servidor [dvorak] a grup [serveis]
2012-01-20 16:12:16	admin	Alta servidor [bartok]
2012-01-20 16:12:16	admin	Alta grup servidor individual [bartok]
2012-01-20 16:12:16	admin	Associat servidor [bartok] a grup [serveis]
2012-01-20 16:12:30	admin	Alta acces grup usuari [B01] a grup servidor [infraestructura]
2012-01-20 16:12:34	admin	Alta acces grup usuari [B02] a grup servidor [serveis]
2012-01-20 16:15:51	admin	Alta usuari [eva]
2012-01-20 16:15:51	admin	Alta grup usuari individual [eva]
2012-01-20 16:16:51	eva	Modificat usuari [eva]
2012-01-20 16:16:51	eva	Desassociat usuari [eva] de tots els grups
2012-01-20 16:16:51	eva	Associat usuari [eva] a grup [B01]

```
48 rows in set (0.00 sec)
```

```
mysql>
```



## 10.2. Script SQL de creació de la DB MySQL

```
SET SQL_MODE="NO_AUTO_VALUE_ON_ZERO";

--
-- Base de dades utilitzada: `ram`
--
CREATE DATABASE IF NOT EXISTS ram;
USE ram;

--
-- Estructura de la taula `Usuaris`
--

DROP TABLE IF EXISTS `Usuaris`;
CREATE TABLE `Usuaris` (
  `username` char(40) NOT NULL,
  `name` char(30) default NULL,
  `surnames` char(60) default NULL,
  `admin` TINYINT(1),
  `info` text,
  `password` char(32),
  `public_key` text NOT NULL,
  PRIMARY KEY (`username`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 ;

--
-- Inserció dades a taula `Usuaris`
--

INSERT INTO `Usuaris` (`username`, `admin`, `public_key`, `password`) VALUES
('admin',1,'00000','9450476b384b32d8ad8b758e76c98a69');

--
-- Estructura de la taula `GrupUsuaris`
--

DROP TABLE IF EXISTS `GrupUsuaris`;
CREATE TABLE `GrupUsuaris` (
  `name` char(20) NOT NULL,
  `info` text,
  `admin` TINYINT(1),
  `individual` TINYINT(1),
  PRIMARY KEY (`name`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 ;

--
-- Estructura de la taula `AgrupacioUsuaris`
--

DROP TABLE IF EXISTS `AgrupacioUsuaris`;
CREATE TABLE `AgrupacioUsuaris` (
  `username` char(40) NOT NULL,
  `groupname` char(20) NOT NULL,
  PRIMARY KEY (`username`,`groupname`),
  FOREIGN KEY (`username`) REFERENCES Usuaris(username),
  FOREIGN KEY (`groupname`) REFERENCES GrupUsuaris(name)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 ;

--
-- Estructura de la taula `Servidors`
--

DROP TABLE IF EXISTS `Servidors`;
CREATE TABLE `Servidors` (
  `hostname` char(40) NOT NULL,
  `ip` char(16) NOT NULL,
  `mixed_mode` TINYINT(1),
  `hash_known_host` char(32),
```

```
    `user` char(10),
    `path` char(30),
    `info` text,
    PRIMARY KEY (`hostname`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 ;

--
-- Estructura de la taula `GrupServidors`
--

DROP TABLE IF EXISTS `GrupServidors`;
CREATE TABLE `GrupServidors` (
  `name` char(20) NOT NULL,
  `info` text,
  `individual` TINYINT(1),
  PRIMARY KEY (`name`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 ;

--
-- Estructura de la taula `AgrupacioServidors`
--

DROP TABLE IF EXISTS `AgrupacioServidors`;
CREATE TABLE `AgrupacioServidors` (
  `hostname` char(40) NOT NULL,
  `groupname` char(20) NOT NULL,
  PRIMARY KEY (`hostname`,`groupname`),
  FOREIGN KEY (`hostname`) REFERENCES Servidors(hostname),
  FOREIGN KEY (`groupname`) REFERENCES GrupServidors(name)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 ;

--
-- Estructura de la taula `Logs`
--

DROP TABLE IF EXISTS `Logs`;
CREATE TABLE `Logs` (
  `date` TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
  `username` char(40) NOT NULL,
  `info` text
) ENGINE=MyISAM DEFAULT CHARSET=utf8 ;

--
-- Estructura de la taula `AgrupacioAccessos`
--

DROP TABLE IF EXISTS `Accessos`;
CREATE TABLE `Accessos` (
  `usergroup` char(20) NOT NULL,
  `servergroup` char(20) NOT NULL,
  `dataexp` DATE,
  PRIMARY KEY (`usergroup`,`servergroup`),
  FOREIGN KEY (`usergroup`) REFERENCES GrupUsuaris(name),
  FOREIGN KEY (`servergroup`) REFERENCES GrupServidors(name)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 ;
```