Manteniment i seguiment de les ofertes públiques d'ocupació

Entrega final

Informació general

Projecte	Oferta pública d'ocupació
Resum	Document d'entrega final del projecte
Autor	Daniel Mas
Data	07/01/09

Història del document

Versió	Descripció	Elaborat per	Data
1.0	Versió inicial	Daniel Mas	7/01/2009

Treball registrat amb una llicència Creative Commons

Contingut del document

Aquest és el document d'entrega final del projecte final de carrera. Aquí hi trobareu una descripció del perquè del projecte, quina era la seva problemàtica a l'hora de dur-lo a terme i l'anàlisi de la seva solució.

El document ens servirà per conèixer a fons el funcionament de l'aplicació, tant a nivell de la tecnologia implementada per dur-lo a terme com del desenvolupament que s'ha fet a partir de les necessitats presents.

Aquest treball està subjecte - excepte que s'indiqui el contrari- en una llicència de Reconeixement-NoComercial-SenseObraDerivada 2.5 Espanya de **Creative Commons**. Podeu copiar-lo, distribuir-los i transmetre'ls públicament sempre que citeu l'autor i l'obra, no es faci un ús comercial i no es faci còpia derivada. La llicència completa es pot consultar en:

http://creativecommons.org/licenses/by-nc-nd/2.5/es/deed.ca

INDEX

Capítol 1 - Introducció **4**
1.1 – Context de l'aplicació 4
1.2 – Objectius de l'aplicació 5
1.3 – Plantejament i mètode seguit 6
1.4 – Planificació del projecte 7
1.5 – Productes obtinguts 8
Capítol 2 - Instal·lant l'entorn **9**
2.1 – Instal·lar Java 9
2.2 – Entorn d'execució (Apache Tomcat + MySQL) 10
2.3 – Instal·lar l'aplicació 11
Capítol 3 - Diagrames dels casos d'ús **12**
3.1 - Actors 12
3.2 – Diagrames de casos d'ús 13
Capítol 4 - Descripció dels casos d'ús **16**
4.1 – Cas d'ús “Validar usuari” 17
4.2 – Cas d'ús “Introduir Publicacions” 18
4.3 – Cas d'ús “Introduir Ofertes” 19
4.4 – Cas d'ús “Consultar Ofertes” 20
4.5 – Cas d'ús “Mantenir Ofertes” 21
4.6 – Cas d'ús “Mantenir Mestre Grups” 22
4.7 – Cas d'ús “Mantenir Mestre Classificacions” 23
4.8 – Cas d'ús “Mantenir Mestre Publicacions” 24
4.9 – Cas d'ús “Introduir usuaris” 25
4.10 – Cas d'ús “Mantenir usuaris” 25
Capítol 5 - Especificació dels casos d'ús **26**
Capítol 6 - Disseny del sistema **31**
6.1 – Definició de les classes model 31
6.2 – Definició de la base de dades 34
Capítol 7 - Disseny del funcionament del sistema **36**
7.1 – Struts i el patró Model-Vista-Controlador 36
7.2 – Estructura del projecte 38
Capítol 8 - Funcionament del sistema **41**
8.1 – Perfil del ciutadà 41
8.2 – Perfil intern 42
8.3 – Cicle de vida d'una oferta 45
Capítol 9 - Conclusions i bibliografia **45**

Capítol 1 - Introducció

1.1 – Context de l'aplicació

Des de fa dos anys estic al departament de desenvolupament d'aplicacions de l'Ajuntament de Terrassa. Darrerament la meva feina s'acostava molt al camp del web, estant algunes de les meves responsabilitats laborals plenament dedicades a algunes aplicacions web, majorment amb gestors de continguts i parcialment, fent el manteniment d'una web feta amb Java Servlets i Java Beans.

Va ser per això que vaig triar el camp del J2EE per fer el meu projecte final de carrera. Això m'ha permès conèixer tecnologies JAVA que no coneixia, com Struts, i aprofundir una mica en el seu desenvolupament.

Després de parlar amb el meu cap, em va posar en contacte amb el departament de personal, que buscaven una aplicació per fer el manteniment i el llistat de les ofertes públiques d'ocupació, és a dir, la publicació de les ofertes laborals de l'Ajuntament.

A l'actualitat, les dades que es mostren a la web es mantenen amb unes aplicacions client-servidor internes de l'Ajuntament, i aquesta aplicació serà la prova per mirar de fer aquest manteniment des d'un entorn web. D'aquesta manera, l'aplicació estarà dividida en dues parts, la part pública que estarà instal·lada en un servidor públic, i la part privada de manteniment de les dades, només accessible internament.

1.2 – Objectius de l'aplicació

La oferta d'ocupació és la via d'entrada per a treballar a l'administració. En ella es publiquen les places per a entrar a treballar a l'Ajuntament, els seus períodes de sol·licitud, publicació i resultats. Aquestes ofertes i les seves actualitzacions han d'estar publicades a la web de l'organisme que les ofereix (en aquest cas, la web de l'Ajuntament de Terrassa) i al Butlletí Oficial de la Província de Barcelona.

L'objectiu de l'aplicació és el de facilitar al departament de personal el manteniment a la web del llistat de les ofertes d'ocupació, i informatitzar aquest procediment que fins ara es feia de manera manual editant manualment fitxers html.

Un cop estigui informatitzat, això facilitarà que s'hi acabin afegint nous mòduls, portar una estadística, un històric, etc.

De la mateixa manera, al ser la primera aplicació que es desenvolupa en la que el manteniment de les dades està en un entorn web, es desenvolupa amb l'objectiu de que el dia de demà n'hi pugui haver més, ubicades en el mateix servidor intern.

És per això que tot i que sigui una aplicació d'ús intern, es fa un control d'identificació dels usuaris, d'aquesta manera s'evita que persones alienes al departament de personal puguin accedir a les dades que aquest publica a internet. Quan en el dia de demà hi hagi més aplicacions al sistema, els usuaris només podran accedir a les aplicacions per les que se'ls hi han assignat privilegis.

Gràcies a aquest projecte he après molt sobre els frameworks java i, concretament, el framework Struts, que permeten estandaritzar força el desenvolupament d'aplicacions java.

1.3 – Plantejament i mètode seguit

La recollida de dades pel projecte s'ha realitzat mitjançant entrevistes amb el departament de personal i una recollida de requeriments, estant en tot moment pendents de com anava avançant el projecte i disposant d'una persona de contacte per si sortia qualsevol dubte en quant al funcionament.
Ha estat molt important i positiu poder disposar dels usuaris reals de l'aplicació a l'hora de treballar en un producte que satisfaci al màxim les seves necessitats.

Un cop fet aquest pas, vaig començar a desenvolupar l'anàlisi de l'aplicació i la base de dades. Degut a circumstàncies personals, la fase de l'anàlisi es va allargar una mica més del previst, i no vaig poder estudiar tant com hagués volgut el framework que havia escollit.

Per tant, la fase de desenvolupament ha estat la més dura de totes, ja que m'he hagut d'enfrontar a la corba d'aprenentatge del framework, als problemes de l'aplicació en si i al poc calendari que m'havia quedat.

Degut a aquesta problemàtica, he hagut d'anar amollant una mica l'aplicació i aplicant prioritats sobre les seves funcionalitats per poder tenir un producte el dia de la entrega. Està previst, però, que un cop feta la entrega, el seu desenvolupament continuï afegint el que hagi descartat en la primera fase i afegint noves funcionalitats que surtin un cop vist el producte final.

1.4 – Planificació del projecte

En un primer moment vaig estimar una programació del projecte mig coincidint amb les dates d'entrega de les proves d'avaluació continuada, però finalment he hagut de forçar aquest calendari al trobar-me que no disposava del temps necessari.

De totes maneres, en la planificació prevista inicialment ja es va treballar amb un marge preveient que això pogués passar, de manera que tot i sortir-me del meu calendari previst inicialment, el producte se n'ha vist menys afectat.

El calendari de les fases ha estat finalment:

Anàlisi i recollida de dades de la situació actual	5 dies (1 - 5 octubre)
Especificacions i funcionalitats a afegir. Disseny aproximat de les interfícies (wireframes):	15 dies (6 - 20 octubre)
Disseny (base de dades, objectes)	15 dies (21 octubre – 5 novembre)
Implementació	50 dies (6 novembre – 26 desembre)
Revisió, elaboració de la documentació i presentació	18 dies (27 desembre – 14 gener)

Es veu a la planificació que la fase d'implementació i revisió són el doble de llargues del que inicialment s'havia pensat, això és així degut a no tenir un correcte aprenentatge de les eines a utilitzar al començar la implementació.

1.5 - Productes obtinguts

El producte final és el d'una aplicació web realitzada amb tecnología J2EE.

Aquesta entrega es composa de:

–Un fitxer zip amb els arxius per al servidor públic aoberta.terrassa.cat i les llibreries per executar el codi de l'aplicació
–Un segon fitxer zip amb els arxius per al servidor intern, que estarà enllaçat des de la intranet municipal.
–El codi font de tota l'aplicació
–Els fitxers de creació de les taules i inserció de les dades bàsiques per a que l'aplicació funcioni
–Instruccions per instal·lar el programari.
–Documentació de la pràctica

El programari desenvolupat és una aplicació amb tecnologia J2EE, per tant es necessitarà un servidor d'aplicacions J2EE per fer-la funcionar i una base de dades MySQL, tot i que és fàcilment migrable a qualsevol tipus de base de dades canviant dos paràmetres i tornant-la a compilar.

Capítol 2 – Instal·lant l'entorn

En aquest capítol s'especifiquen les instruccions per instal·lar l'entorn d'execució del projecte. Si ja es disposa d'un entorn Tomcat i MySQL, es pot obviar aquest capítol.

Els arxius executables no estan inclosos amb la pràctica, però són tots gratuïts i es poden descarregar lliurement dels enllaços que especifico.

El programari emprat en la execució de l'aplicació és:

•Java versió 1.5.0_13
•Apache Tomcat 5.5 com a servidor web
•MySQL 5.0 com a servidor de base de dades
•Apache Struts com a tecnologia java utilitzada
•Eclipse 3.4 com a IDE

Recordem que un cop estigui en producció, la aplicació s'executarà en dos servidors tomcat diferents:

•Un on hi haurà la part interna i de gestió de les dades de l'aplicació (un backoffice) i
•Un altre ja en funcionament on s'hi implementaran les pàgines públiques de l'aplicació.

Per facilitar la implementació la estic realitzant tota sobre el mateix tomcat però amb dues carpetes diferents: a la carpeta /*personal*, tota la part interna, i a la carpeta /*ofertes*, tota la part externa..

2.1 - Instal·lar Java

Des de la web de Java ens descarreguem el paquet de java 1.5 sense Netbeans (no ens cal):

http://java.sun.com/javase/downloads/index_jdk5.jsp

Un cop descarregat i instal·lat, haurem de crear la variable d'entorn JAVA_HOME que apunti cap a la carpeta on ho hem instal·lat, per exemple

C:\Archivos de programa\Java\jdk1.5.0_13

Després afegim a la variable *PATH* el valor *%JAVA_HOME%/bin* i mirem si existeix la variable *CLASSPATH*, si existeix li afegim al final el valor `;.` i ja està.

2.2 - Entorn d'execució (Apache Tomcat + MySQL)

Apache Tomcat

Ens descarreguem el Tomcat 5.5 des de la seva pàgina i l'instal·lem en una carpeta del nostre PC, per exemple a c:\tomcat:

http://tomcat.apache.org/download-55.cgi

Per configurar el Tomcat, obrim l'arxiu server.xml que està dins de la carpeta /conf i afegim aquestes línies just abans de *</Host>*:

```
<Context path="/personal"
docBase="C:\workspace\Struts\WebContent"
reloadable="true" />
```

D'aquesta manera hem configurat el tomcat perquè si hi accedim així http://localhost:8088/personal/ accedirem a l'arrel de la nostra aplicació, per tant ja teníem el nostre context creat.

MySQL

Fem el mateix amb el MySQL, descarregant-lo des de la seva pàgina. La pràctica s'ha desenvolupat sobre MySQL 5, però pot funcionar sobre qualsevol versió:

http://dev.mysql.com/downloads/

Al MySQL mentre estic desenvolupant estic utilitzant una base de dades anomenada new_bd (nom per defecte).

Per tant, haurem de crear una base de dades amb aquest nom i tot seguit, executar les ordres del fitxer **new_db.sql** inclòs en la entrega.

En les meves proves utilitzo l'usuari root per connectar amb la base de dades amb la contrasenya personal però un cop acabi l'aplicació aquestes dades seran configurades.

2.3 - Instal·lar l'aplicació

En una carpeta local:

Descomprimim el .rar dins de la carpeta c:\workspace, i veurem com s'ha creat una carpeta /Struts on al seu interior hi ha la carpeta WebContent que conté els fitxers jsp, i les classes jsp.

Un cop tenim la nostra aplicació descomprimida, amb el Tomcat i el MySQL funcionant, només haurem de posar al nostre navegador l'adreça que hem esmentat abans http://localhost:8088/uoc/

Al servidor públic:

Ens connectem al servidor de l'administració oberta i pugem la carpeta /ofertes i l'arxiu /css/publica.css a la carpeta /css.

Pugem també els arxius de les llibreries Struts a */WEB-INF/lib* i els fitxers que comencen per struts a */WEB-INF/* i la carpeta */WEB-INF/classes/org/terrassa/uoc/* dins de la mateixa carpeta al servidor.

Usuaris de l'aplicació

Un cop instal·lat el sistema es recomana identificar-se amb l'usuari administrador del sistema, donar d'alta nous usuaris i canviar la clau de l'administrador.

L'identificador de l'administrador i la seva clau són "*admin*" i "*admin*", respectivament.

Capítol 3 – Diagrames de casos d'ús

3.1 - Actors

Tota aplicació s'executa dins d'un escenari. Aquest escenari està composat de personatges i accions. Els personatges del nostre escenari són els actors, i les accions els casos d'ús.

Anem a veure una breu descripció dels actors que intervenen en la nostra aplicació

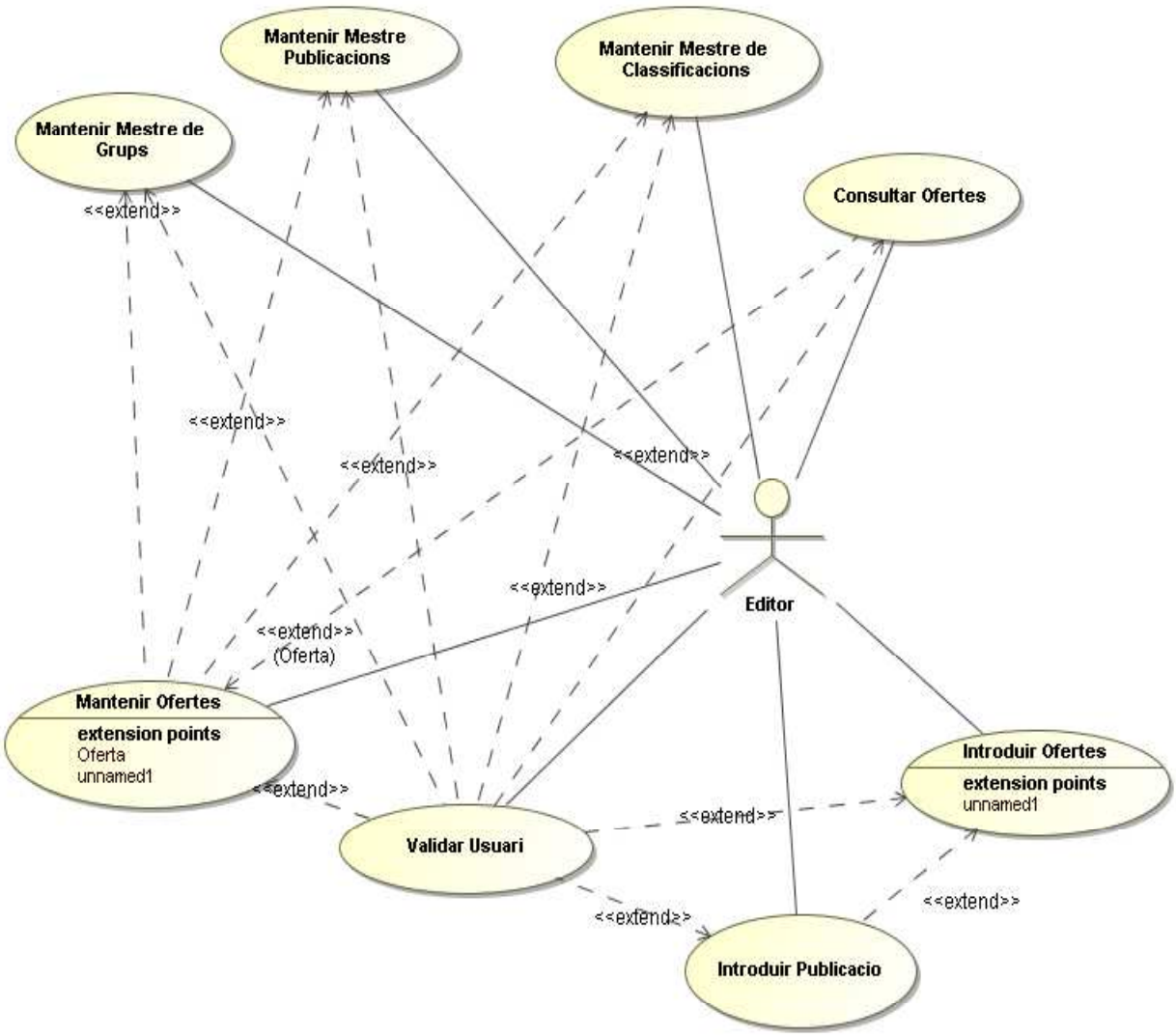
Actor	**Descripció**	**Responsabilitats**
Anònim	És el navegant anònim d'internet. El ciutadà que entrarà al nostre portal per consultar les dades de les ofertes d'ocupació.	No té cap responsabilitat, en si és el client de la nostra aplicació, i ha de poder trobar la informació que busca de la manera més còmoda i ràpida possible.
Editor	És el membre o els membres del departament de personal que s'encarregaran de mantenir les dades de les ofertes publicades.	Són els responsables de les dades publicades per l'aplicació, ja que són els qui generen les ofertes d'ocupació, les seves convocatòries i els seus documents, i els hauran d'adjuntar amb la seva respectiva oferta perquè surtin reflectits a internet i a disposició de consulta pel ciutadà.
Administrador	És l'administrador que s'encarrega de mantenir les dades de la nostra aplicació	És el responsable de les dades de l'aplicació que utilitzarà l'usuari Editor. També és qui s'encarregarà de donar d'alta als diferents usuaris de l'aplicació.

3.2 - Diagrames de casos d'ús

Anem a veure ara les accions que haurà de fer la nostra aplicació. Aquestes accions s'anomenen casos d'ús, i aquí els veiem agrupats pels actors que els executaran.

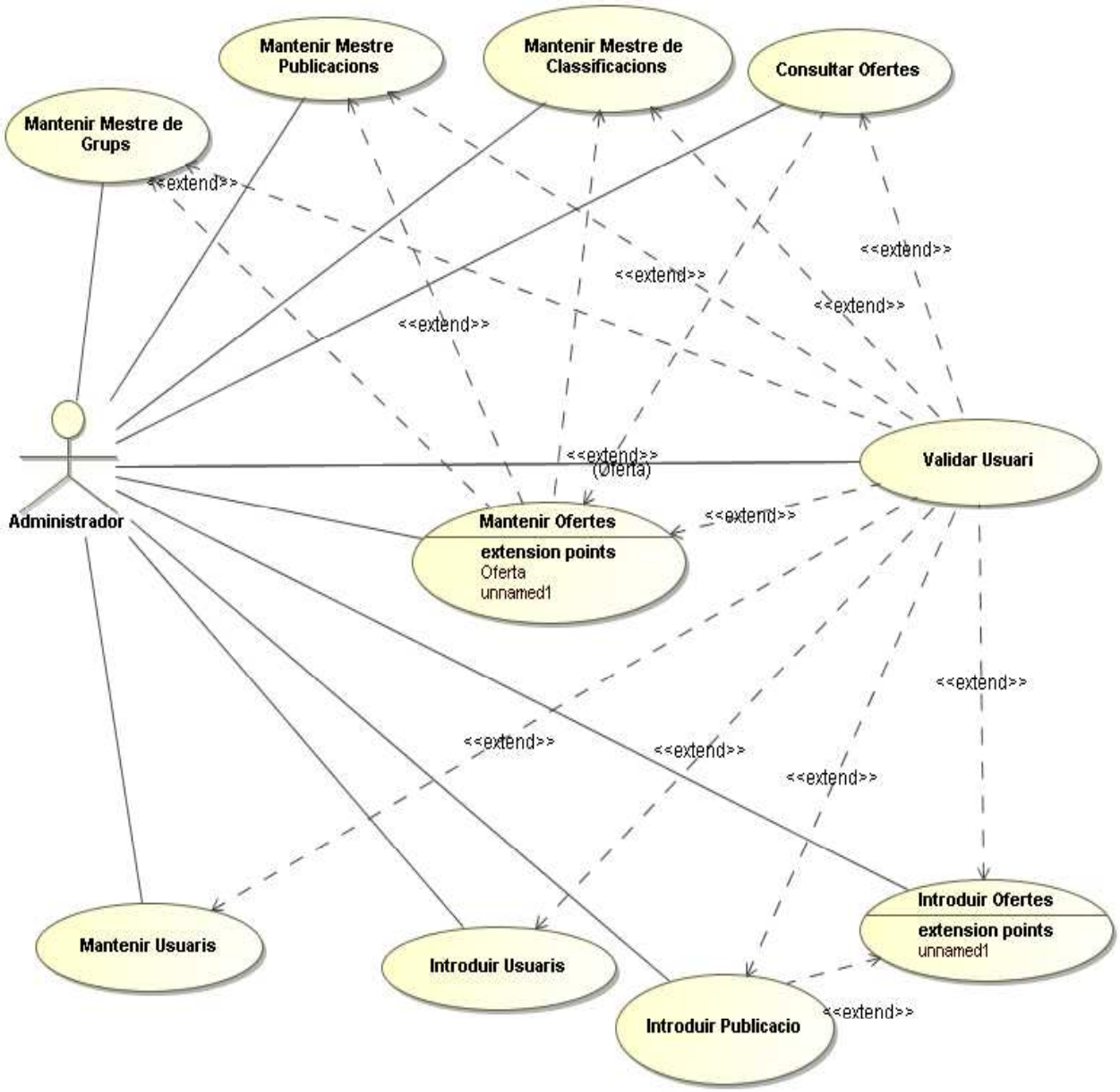
Casos d'ús de l'actor Editor

Veiem que les accions que directament executarà l'Editor són principalment les de introducció i manteniment de les dades de l'aplicació, i aquestes dependran d'haver passat anteriorment per l'acció de validar usuari.



Casos d'ús de l'actor Administrador

L'actor Administrador podrà fer també totes les accions que fa l'usuari Editor, i com a afegit també pot crear nous usuaris de l'aplicació, i encarregar-se del seu manteniment.



Casos d'ús de l'actor Anònim

L'actor anònim simplement farà consultes de les ofertes donades d'alta al sistema des de la seva part pública.

Capítol 4 - Descripció dels Casos d'ús

En aquest apartat s'explicarà amb més detall el contingut dels diferents casos d'ús. En ells s'explicarà:

Resum: Petit resum del cas d'ús

Paper: Quin paper o funció fa el cas d'ús dins de l'aplicació

Actors: Quins **actors** estan implicats en el cas d'ús

Casos d'ús relacionats: Amb quins altres casos d'ús té algun tipus de relació

Precondició: Quina condició s'ha de donar perquè s'executi el cas d'ús, si n'hi ha.

Postcondició: Resultat final quan el cas d'ús ha acabat, si n'hi ha.
Breu explicació del cas d'us i altres detalls

Classes implicades: Classes d'entitats que identifiquem en el cas d'ús.

Flux / Subfluxos: Flux de dades i del procediment d'execució del cas d'ús

Fluxos alternatius: Flux del procediment en cas de que aquest es trobi amb algun problema en la seva execució.

4.1 - Cas d'ús “Validar Usuari”

Resum: Un usuari s'identifica al sistema.

Paper: Els usuaris introdueixen un nom d'usuari i una contrasenya. El cas d'ús identifica quin tipus d'usuari és i li mostra les seves opcions disponibles.

Actors: Administrador, **Editor**. Tots els usuaris que es puguin identificar correctament al sistema.

Casos d'ús relacionats: Mantenir Ofertes, Consultar Ofertes, Introduir Ofertes, Introduir Usuaris, Mantenir Usuaris.

Precondició: Els usuaris han d'estar donats d'alta al sistema o el procés d'identificació no es portarà a terme.

Postcondició: Els usuaris ja estan identificats al sistema i poden utilitzar les opcions del menú que tenen assignades.

A la sortida d'aquest cas d'ús els actors ja tindran el seu rol identificat, ja sigui com a **Editor** o com a **Administrador**.
Serà el cas d'ús que primer s'haurà d'executar al entrar al portal.

Classes implicades: Aplicació, Usuari

Flux

1. L'usuari entra a la pantalla d'identificació
2. Omple les dades de nom d'usuari i contrasenya, proporcionades anteriorment per un administrador
3. El sistema comprova la identificació de l'usuari
4. Un cop ha comprovat que existeix i la identificació ha estat correcta, el sistema mostra les opcions de menú de l'usuari.

Fluxos alternatius:

Usuari no trobat

1. El sistema troba que l'usuari no existeix o que la contrasenya no és correcte
2. Ens mostra una pantalla d'error on podem tornar a identificar-nos o també ens suggereix posar-nos en contacte amb un administrador.

Diagrama de seqüències: Veure Diagrama 1 (pag. 14)

4.2 - Cas d'ús “Introduir Publicacions”

Resum: Dóna d'alta una publicació d'ofertes al sistema

Paper: És el cas d'ús en que es donen d'alta les publicacions on es publiquen les diferents ofertes al ser creades.

Actors: Editor. Anomenarem així a la persona del departament de personal encarregada del manteniment de l'aplicació.

Casos d'ús relacionats: Mantenir Mestre de Publicacions, Introduir Ofertes, Mantenir Ofertes

Precondició: El mitjà de publicació ha d'existir al mestre de les publicacions, la oferta ha d'estar donada d'alta al sistema.

L'**editor**, un cop s'ha identificat al sistema, accedeix a la opció de donar d'alta una nova publicació. Un cop es dona d'alta la publicació indicarem les dades de la publicació i quines ofertes s'han publicat.

Classes implicades: Oferta, Publicacions, Mitjans

Flux

1. L'usuari selecciona la opció del menú de crear una nova publicació
2. Apareix la pantalla d'afegir publicacions, on l'usuari selecciona una publicació de les disponibles i introdueix el número de publicació o butlletí i la data de la publicació.
3. L'usuari selecciona d'una llista de les ofertes introduïdes les publicades en aquesta publicació i al guardar els canvis aquestes apareixen en un llistat d'ofertes introduïdes.

Fluxos alternatius:

1. La oferta publicada no està donada d'alta
2. Hi ha un botó d'afegir ofertes que comunicarà directament amb el cas d'ús “Introduir Ofertes”.
3. Un cop donada d'alta, l'usuari ja la pot afegir a la publicació

4.3 - Cas d'ús “Introduir Ofertes”

Resum: Dóna d'alta una nova oferta d'ocupació al sistema

Paper: És el cas d'ús principal que dóna d'alta els registres que anirem actualitzant i de que principalment es nodreix l'aplicació.

Actors: Editor. Anomenarem així a la persona del departament de personal encarregada del manteniment de l'aplicació.

Casos d'ús relacionats: Mantenir Ofertes, Consultar Ofertes

Precondició: La oferta no ha d'existir prèviament i l'usuari ha de ser **editor**.

Postcondició: La oferta ja existeix a la base de dades.

L'**editor**, un cop s'ha identificat al sistema, accedeix a la opció de donar d'alta una nova oferta de formació, introduint les dades principals d'aquesta de que disposa.

Alternatives de procés i excepcions: L'**editor** pot fer aquesta activitat directament o havent fet prèviament una consulta de les ofertes existents mitjançant Consultar Ofertes. Després d'aquesta consulta i que l'**editor** hagi comprovat que no existeix anteriorment, procedirà a donar-la d'alta.

Classes implicades: Oferta

Flux

1. L'usuari selecciona la opció d'alta d'oferta
2. S'obre un formulari perquè l'usuari introdueixi les dades de la oferta amb els camps obligatoris senyalats
3. Després d'haver omplert les dades, al pitjar el botó guardar les dades ja estan donades d'alta.

Fluxos alternatius:

No s'omplen tots els camps obligatoris

1. L'usuari pitja el botó guardar i no hi ha tots els camps obligatoris omplerts
2. Apareix una altra vegada el formulari amb les dades i un missatge que adverteix a l'usuari de la situació i li diu quin camp obligatori s'ha deixat
3. Si l'usuari omple les dades que falten, al apretar el botó guardar la oferta es donarà d'alta. Mentre no ompli les dades, el procés es quedarà tota la estona en el pas 2.

Diagrama de seqüències: Veure Diagrama 2 (pag. 15)

4.4 - Cas d'ús “Consultar Ofertes”

Resum: Retorna un llistat amb les ofertes que hi ha al sistema

Paper: Com ens retorna un llistat, és un cas d'ús que cridarem des de diferents casos d'ús i que necessitarem quan haguem de fer una selecció entre les ofertes existents.

Actors: Editor, Anònim (anomenarem així al navegant que entra a la web per consultar-hi la informació, sense passar per la validació d'usuaris).

Casos d'ús relacionats: Mantenir Ofertes.

Precondició: L'usuari pot ser editor o anònim.

Postcondició: Ens retorna un llistat de les ofertes de la base de dades

Tot i que el resultat en temes de contingut és pràcticament el mateix (una llista de les ofertes) hem de tenir en compte que la seva aparença gràfica serà diferent depenent de amb quin usuari cridem a la consulta, ja que un ho farà des de l'entorn d'administració i l'altre des de l'entorn públic.
Les ofertes poden aparèixer ordenades per data, per classificació i per publicació.
La pantalla de consulta de l'**editor** pot tenir dos possibles formats, la del llistat de les ofertes en que apareixeran individualment una sota l'altra i la del llistat de les publicacions, en que apareixeran les diferents publicacions i les ofertes que hi surten publicades.

Classes implicades: Oferta, Publicacions

Flux

Seleccionar ofertes segons publicació

1. L'usuari entra a la pantalla de consulta d'ofertes.
2. Apareix un llistat de les publicacions fetes fins ara.
3. Al seleccionar l'usuari una de les publicacions, s'obre una pàgina amb un llistat de les ofertes publicades en aquesta publicació.
4. Les ofertes són enllaços, per tant si l'editor selecciona alguna oferta passarà directament a la pantalla i cas d'ús de “Mantenir Ofertes”

La oferta encara no s'ha publicat però està donada d'alta al sistema per planificació

1. A la pantalla de consulta hi ha també un formulari de cerca on l'usuari pot introduir els seus criteris per fer la consulta:
   1. Grup
   2. Any publicació
   3. Fase en la que estan les ofertes (publicada, pendent, en tràmit, etc...)
2. L'usuari pitja el botó de buscar i el sistema li retorna una llista que correspon al/s criteri/s de cerca que hagi seleccionat.

Diagrama de seqüències: Veure Diagrama 3 (pag. 16)

4.5 - Cas d'ús “Mantenir Ofertes”

Resum: Fem un manteniment de l'estat de les ofertes.

Paper: En aquest manteniment es van actualitzant les dades de les ofertes a mida que el procés va avançant, s'hi afegeixen els plaços i les actes i resolucions.

Actors: Editor.

Casos d'ús relacionats: Consultar Ofertes, Mantenir Grups, Mantenir Classificacions.

Precondició: La oferta ha d'estar donada d'alta prèviament.

En aquest manteniment és on es centra l'activitat més important de l'aplicació, ja que és a les dades que en ell s'actualitzen a les que els usuaris volen accedir. És on finalment els editors hi concentraran una major quantitat de treball, per tant interessa que sigui fàcil de mantenir i utilitzar.

Classes implicades: Oferta, Grup, Classificació

Flux

1. L'usuari ha arribat al manteniment de les ofertes després de la crida al cas d'ús “Consultar Ofertes”
2. Es dibuixa a la pantalla un formulari omplert amb les dades de la oferta. Així, si es vol modificar alguna dada es pot fer directament sobre el camp.
3. L'usuari modifica les dades que desitja
4. Al apretar el botó de guardar dades, la oferta està actualitzada

Diagrama de seqüències: Veure Diagrama 4 (pag, 17)

4.6 - Cas d'ús “Mantenir Mestre Grups”

Resum: Fem un manteniment dels grups de les ofertes.

Paper: Aquests grups són els diferents tipus de plaça. No es canvien freqüentment, ja que l'última classificació que s'ha fet és la establerta en L'Estatut Bàsic de l'Empleat Públic (Llei 7/2007 de 12 d'abril). Es basen en la formació necessària per dur-la a terme.

Actors: Editor.

Casos d'ús relacionats: Consultar Ofertes, Mantenir Ofertes, Introduir Ofertes

Aquest cas d'ús agrupa tot el manteniment dels grups de les ofertes públiques d'ocupació. Es podran fer altes de nous grups i fins i tot editar els existents per si hi hagués algun canvi a la llei que obligués a l'administració a fer una nova classificació.
Actualment, aquesta classificació és: A1, A2, B, C1, C2, APG.

Classes implicades: Oferta, Grup

Subfluxos:

Alta de grup
1. L'usuari selecciona la opció d'alta
2. Apareix el formulari per introduir el nom del grup
3. S'introdueixen les dades i s'apreta el botó guardar.
4. Les dades del grup estan guardades

Baixa de grup
1. L'usuari selecciona la opció de baixa del grup
2. El sistema demana confirmació
3. La baixa està realitzada

Modificació de grup
1. L'usuari selecciona el grup de la llista
2. Apareix el formulari amb les dades del grup
3. L'usuari les modifica directament i guarda el grup

Fluxos alternatius:

El nom del grup ja existeix (al fer l'alta o modificació)
1. El sistema retorna un missatge a l'usuari en que li diu que aquest grup ja existeix, i en el missatge li permet seleccionar l'existent.
2. Si el selecciona, s'obre la finestra de modificar aquest grup

Hi ha ofertes d'aquest grup (al fer baixes)
1. El sistema avisa a l'usuari de que si hi ha alguna oferta que té marcat aquest grup no es pot fer la baixa, i mostra una llista de les ofertes que compleixen aquesta condició.
2. L'usuari edita les ofertes i canvia el grup que tenen
3. Un cop fet això, pot tornar al pas 1 de la baixa.

4.7 - Cas d'ús “Mantenir Mestre Classificacions”

Resum: Fem un manteniment de les classificacions

Paper: Aquesta classificació es basa en la funció que realitzarà qui estigui ocupant la plaça. Es basa en la escala i el servei. Tampoc tenen una actualització molt freqüent.

Actors: Editor.

Casos d'ús relacionats: Consultar Ofertes, Mantenir Ofertes, Introduir Ofertes

Aquest cas d'ús agrupa tot el manteniment de les classificacions segons escala i servei de les diferents ofertes. Es podran fer altes de noves classificacions i fins i tot editar les existents per si hi hagués algun canvi a la llei que obligués a l'administració a fer una nova classificació.

Classes implicades: Oferta, Classificació

Subfluxos:

Alta de classificació

1. L'usuari selecciona la opció d'alta
2. Apareix el formulari per introduir el nom de la classificació
3. S'introdueixen les dades i s'apreta el botó guardar.
4. Les dades de la classificació estan guardades

Baixa de classificació

1. L'usuari selecciona la opció de baixa de la classificació
2. El sistema demana confirmació
3. La baixa està realitzada

Modificació de classificació

1. L'usuari selecciona la classificació de la llista
2. Apareix el formulari amb les seves dades
3. L'usuari les modifica directament i guarda la classificació

Fluxos alternatius:

El nom de la classificació ja existeix (al fer l'alta o modificació)

1. El sistema retorna un missatge a l'usuari en que li diu que ja existeix un registre anomenat així, i li mostra el nom del registre.
2. L'usuari pot seleccionar-lo si vol perquè s'obri la finestra de modificar aquest registre.

Hi ha ofertes d'aquesta classificació (al fer baixes)

1. El sistema avisa a l'usuari de que si hi ha alguna oferta que té marcada aquesta classificació no es pot fer la baixa, i mostra una llista de les ofertes que compleixen aquesta condició.
2. L'usuari edita les ofertes i canvia la seva classificació.
3. Un cop fet això, pot tornar al pas 1 de la baixa.

4.8 - Cas d'ús “Mantenir Mestre Publicacions”

Resum: Fem un manteniment dels tipus de publicacions

Paper: En un principi només hi haurà els butlletins de publicació, BOE, BOP, DOGC, Gaceta Municipal.

Actors: Editor.

Casos d'ús relacionats: Consultar Ofertes, Mantenir Ofertes, Introduir Ofertes

Aquest cas d'ús agrupa el manteniment dels mitjans de publicacions en que apareixeran les ofertes. Només es donen d'alta els diaris i butlletins, no els diferents números de les publicacions. Per donar d'alta els diferents números això ja es fa al Introduir Publicació.

Classes implicades: Oferta, Mitjans

Subfluxos:

Alta de mitjà

1. L'usuari selecciona la opció d'alta
2. Apareix el formulari per introduir el nom del mitjà
3. S'introdueixen les dades i s'apreta el botó guardar.
4. Les dades del mitjà estan guardades

Baixa de mitjà

1. L'usuari selecciona la opció de baixa del mitjà
2. El sistema demana confirmació
3. La baixa està realitzada

Modificació del mitjà

1. L'usuari selecciona el mitjà de la llista
2. Apareix el formulari amb les seves dades
3. L'usuari les modifica directament i guarda el mitjà

Fluxos alternatius:

Hi ha números d'aquesta publicació (al fer baixes)

1. El sistema avisa a l'usuari de que si hi ha alguna publicació d'aquest mitjà no es pot fer la baixa, i mostra una llista de les publicacions que compleixen la condició.
2. O s'esborren els números del mitjà o no es podrà fer la baixa.

4.9 - Cas d'ús “Introduir Usuaris”

Resum: Donem d'alta als usuaris al sistema.

Paper: En aquest cas d'ús assignem a un usuari un nom, una contrasenya i un rol perquè entrin al sistema. Es fa una alta al sistema amb les dades de l'usuari.

Actors: **Administrador.**

Casos d'ús relacionats: Mantenir Usuaris, Validar Usuari

L'administrador s'encarrega de donar d'alta als nous usuaris al sistema mitjançant aquest sistema, i els hi assigna la seva funció com a administrador o com a editor de la aplicació.
En aquesta aplicació només n'hi haurà una, però en un futur pot ser que el nombre augmenti.
Classes implicades: Usuari, Aplicació

Diagrama de seqüències: Veure Diagrama 5 (pag, 18)

4.10 - Cas d'ús “Mantenir Usuaris”

Resum: Gestionem els usuaris del sistema

Paper: Veiem els usuaris que hi ha donats d'alta i els editem, podent canviar les seves dades: nom, funció, contrasenya, etc.

Precondició: L'usuari ha d'estar ja donat d'alta al sistema.

Actors: **Administrador.**

Casos d'ús relacionats: Introduir Usuaris, Validar Usuari

Classes implicades: Usuari, Aplicació

Capítol 5 – Especificació dels casos d'ús

Tot seguit es presentaran els diagrames de seqüències dels casos d'ús més destacables de l'aplicació. Al apartat anterior es pot trobar amb més detall la descripció del que fa cada cas d'ús.

S'han obviat els diagrames de seqüències dels casos d'ús dels manteniments (ja anomenats Manteniment xxx) ja que tot i contenir les operacions d'alta, baixa i modificació, són operacions molt simples amb només una taula implicada, i tots molt semblants entre ells.

Diagrama 1 – Validar Usuari

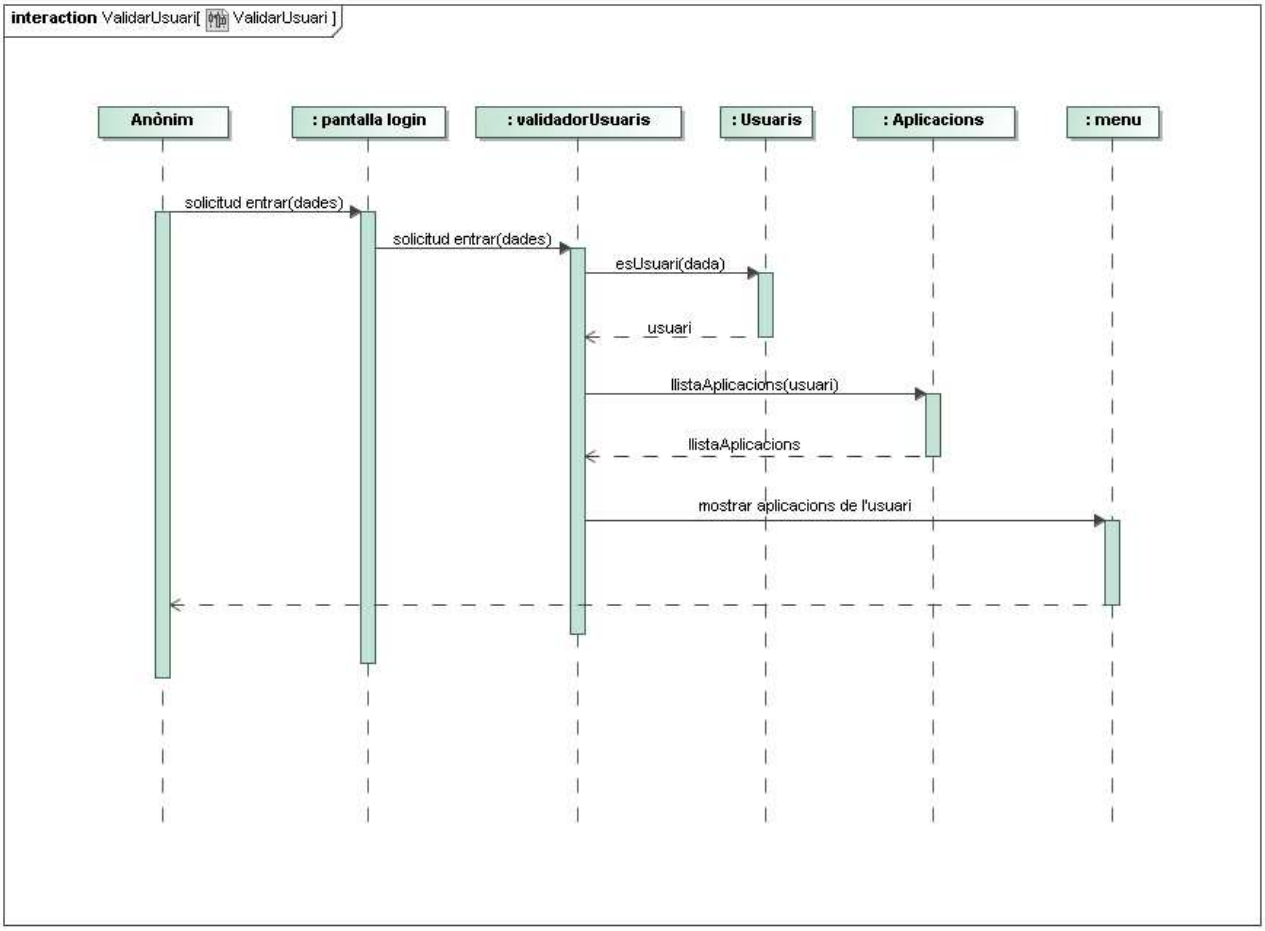


Diagrama 2 – Introduir Ofertes

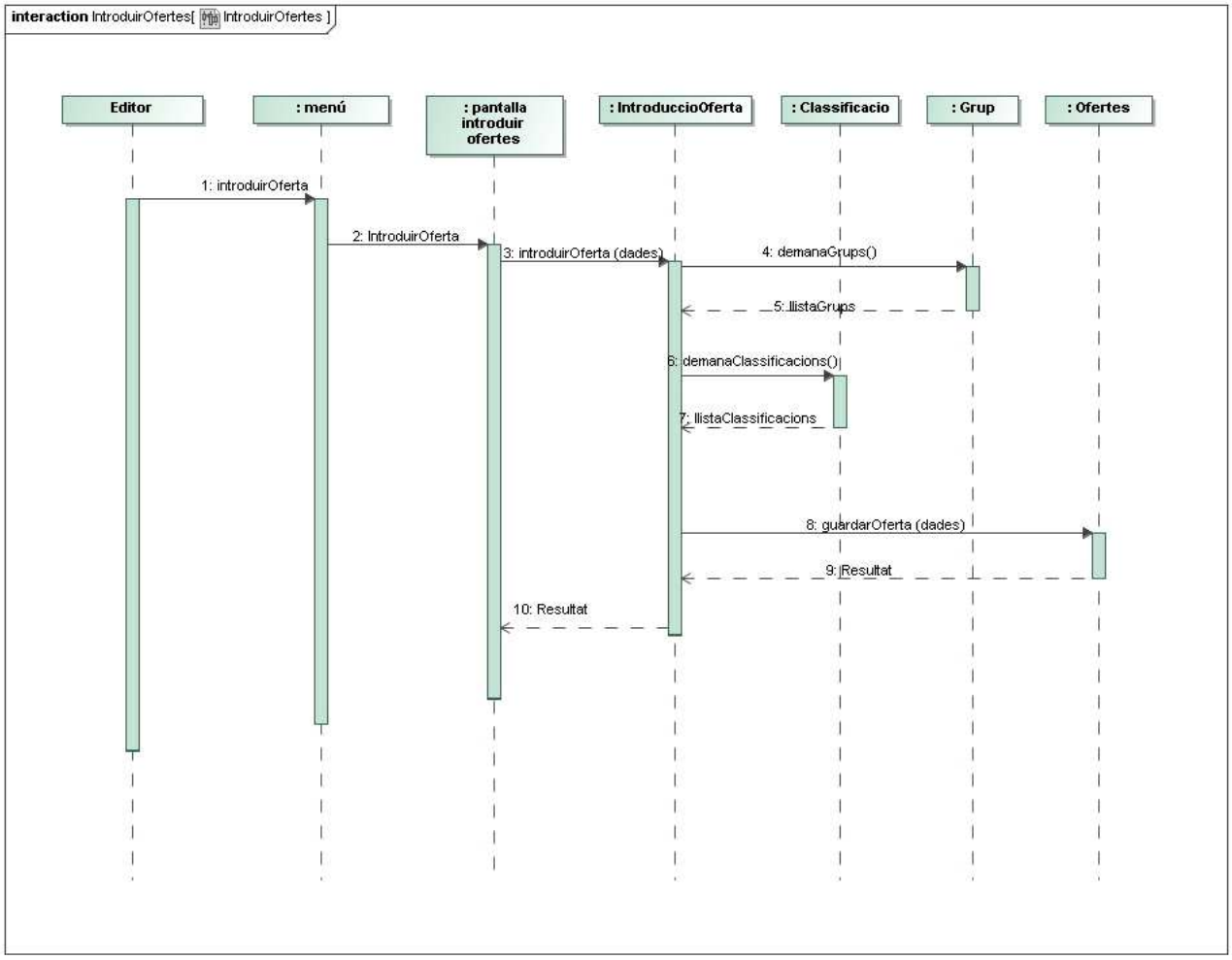


Diagrama 3 – Consultar Ofertes

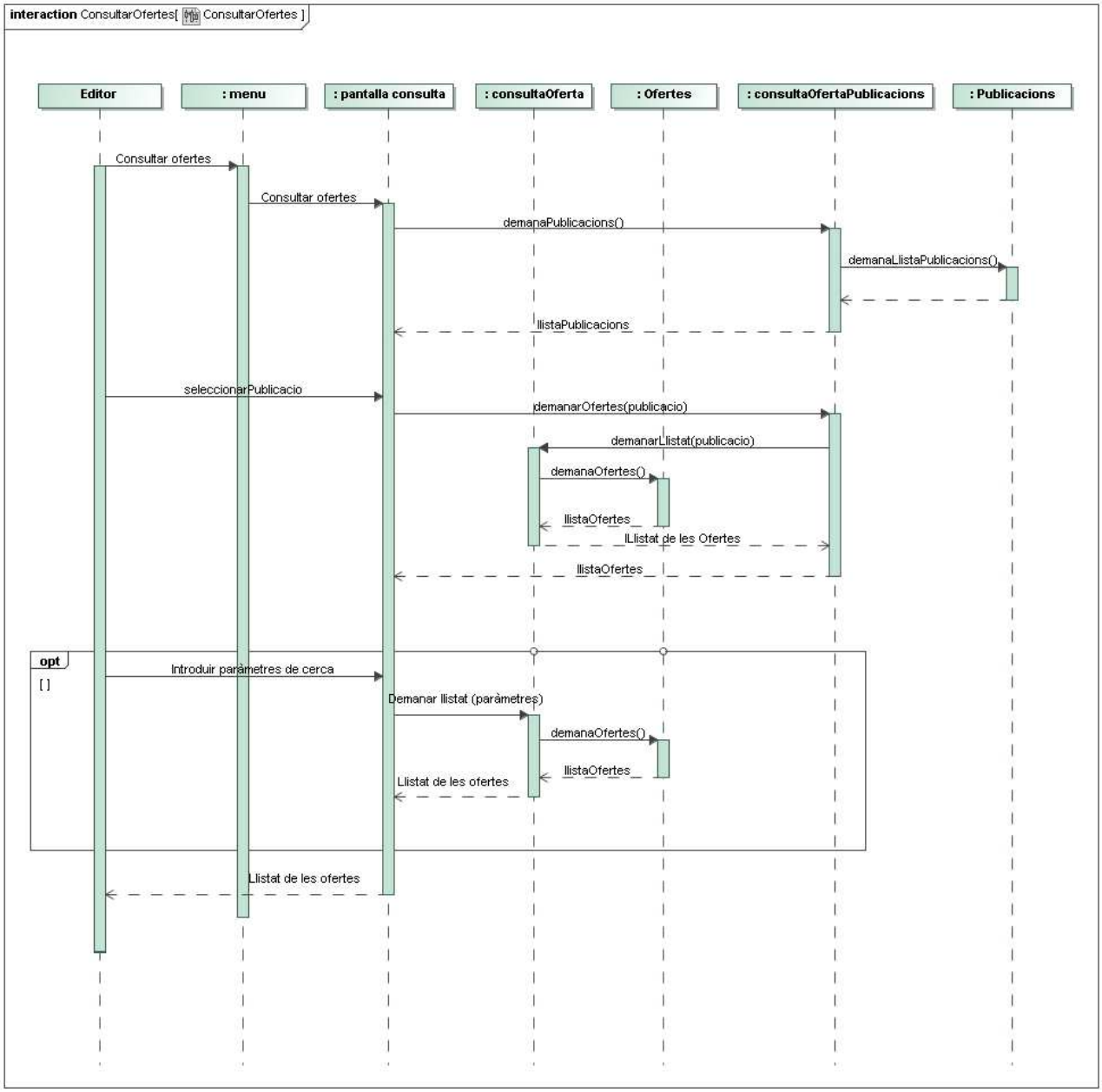


Diagrama 4 – Mantenir Ofertes

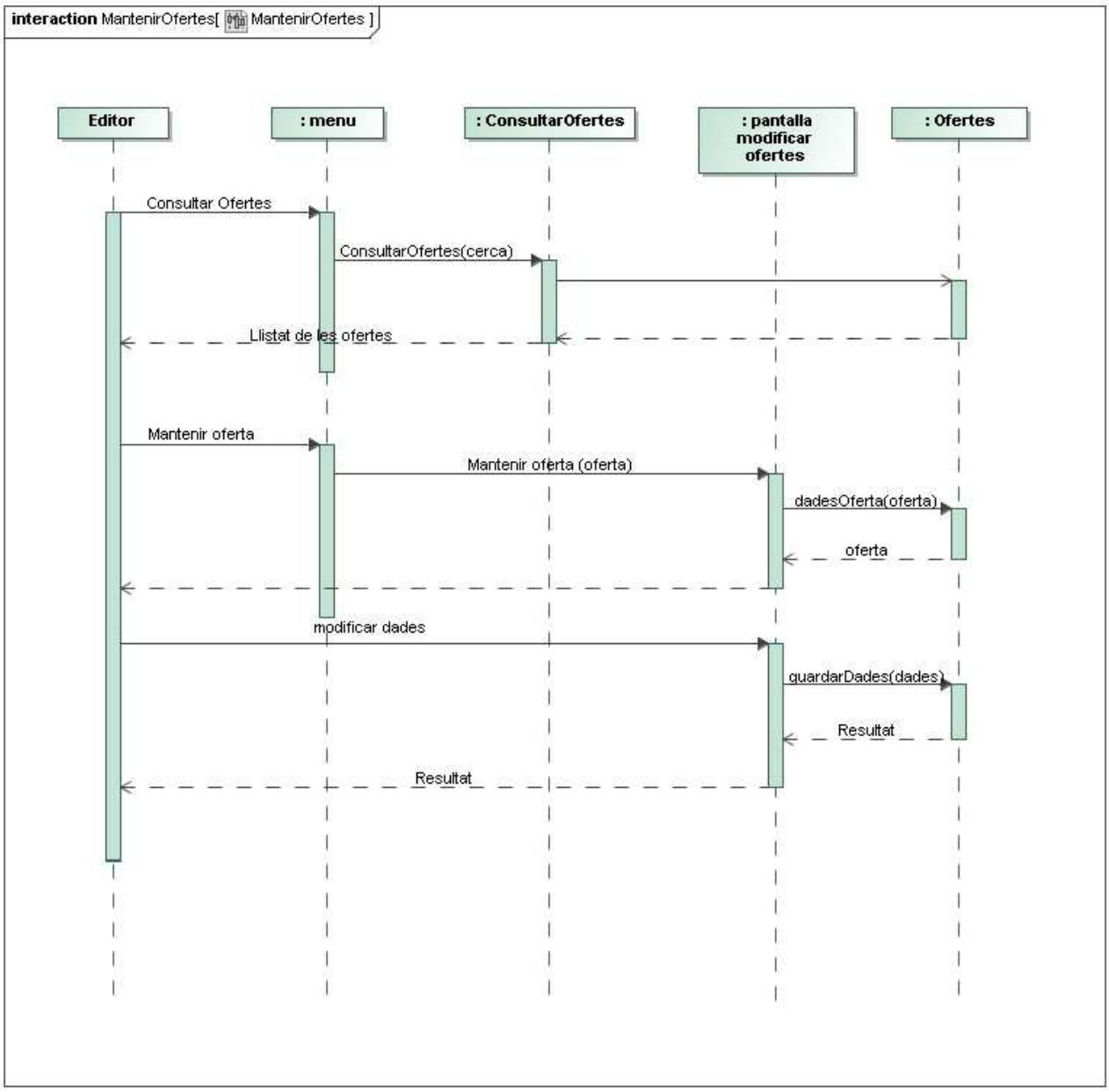
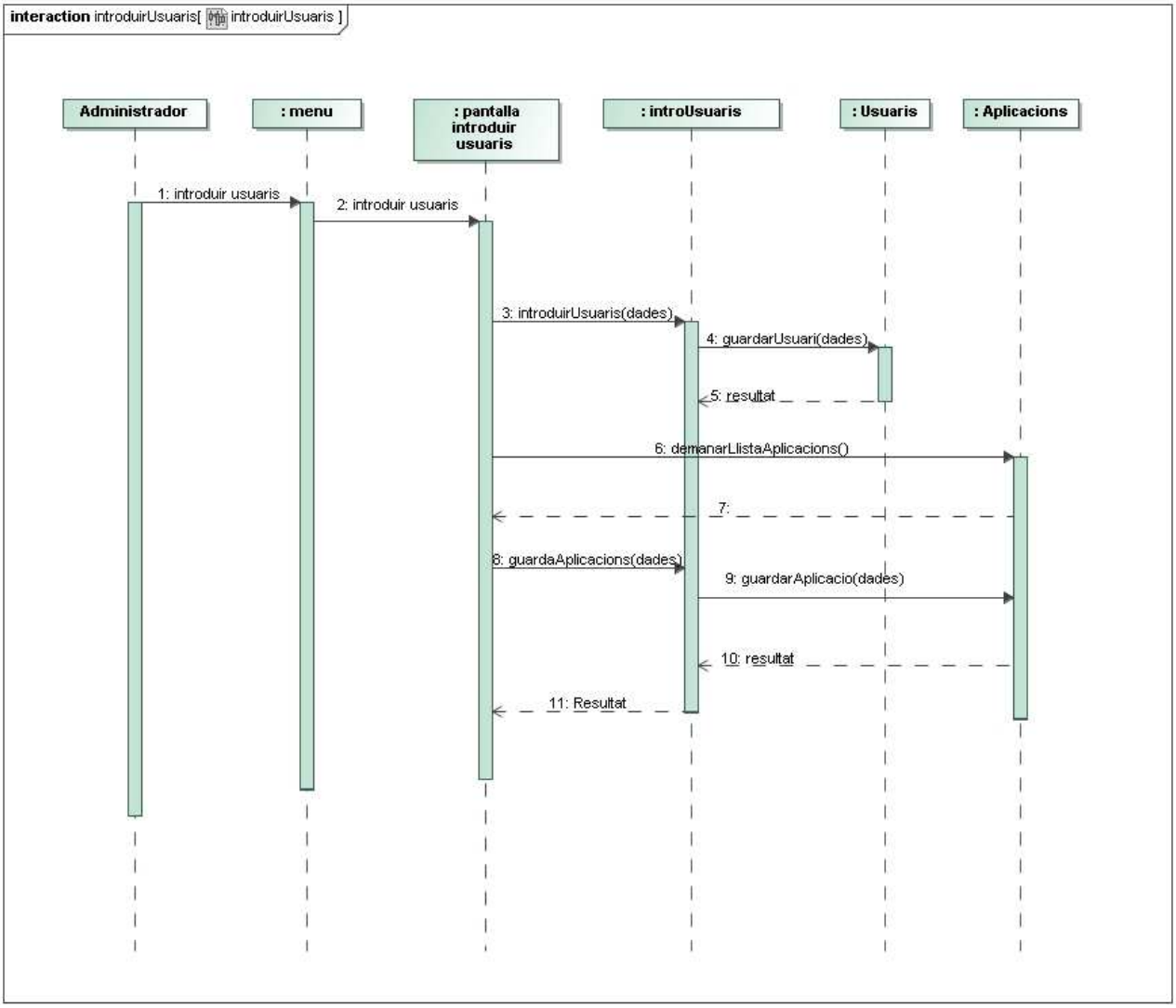


Diagrama 5 – Introduir Usuaris



Capítol 6 - Disseny del sistema

6.1 - Definició de les classes model

Anem a definir detalladament les classes que hem vist implicades en l'aplicació gràcies als casos d'ús, i els hi anem a definir els seus atributs.

Classe Aplicacions
- idAplicació (*integer*)
- nom (*string*): Nom de l'aplicació que es veurà als menús
- descripció (*string*): Descripció de l'aplicació
- ruta (*string*): Carpeta dins del servidor on estaran els arxius de l'aplicació

Classe Usuaris
- idUsuari (*integer*)
- nom (*string*): Nom de l'usuari
- cognoms (*string*): Cognoms de l'usuari
- nlogin (*string*): Nom d'usuari per identificar-se al sistema
- npass (*string*): Contrasenya de l'usuari
- area (*string*): Àrea del departament de l'usuari, merament informatiu

Classe Ofertes
- idOferta (*integer*)
- idGrup (*integer*)
- idClassificacio(*integer*)
- nomOferta (*string*): Nom de la oferta o lloc de treball
- bases (*arxiu*): Arxiu en format PDF amb les bases publicades al BOP
- nPlaces (*integer*): Número de places convocades
- sistemaSeleccio (*string*): Sistema de selecció de la plaça, pot ser concurs o concurs/oposició
- dataPublicació (*data*): Data de la publicació al BOP, també data en la que apareixerà a la web
- dataTermini (*data*): Data final per la presentació de sol·licituds
- resolució (*arxiu*): Arxiu en format PDF amb la resolució dels admesos i exclosos per al procés
- dataProves (*data*): Data en la que es realitzaran les proves, si n'hi ha
- horaProves (*hora*): Hora de realització de les proves
- llocProves (*string*): Lloc on es faran les proves
- acta (*arxiu*): Acta final sobre el procés
- usuModif (*integer*): Darrer usuari en modificar o donar d'alta aquesta oferta
- dataModif (*data*): Data en la que s'ha fet l'última modificació o alta d'aquesta oferta.

Classe Classificació
- idClassificacio (*integer*)
- descripcio (*string*)

Classe Grup
- idGrup (*integer*)
- descripcio (*string*)

Classe Mitjans
- idMitja (*integer*)
- nom (*string*): Inicials del nom del mitjà, normalment s'anomena per les inicials.
- descripcio (*string*): Descripció del significat de les inicials.

Classe Publicacions
- idMitja (*integer*)
- numPublicacio (*integer*): Número de la publicació en que s'ha publicat la oferta
- dataPublicacio (*data*): Data de la publicació

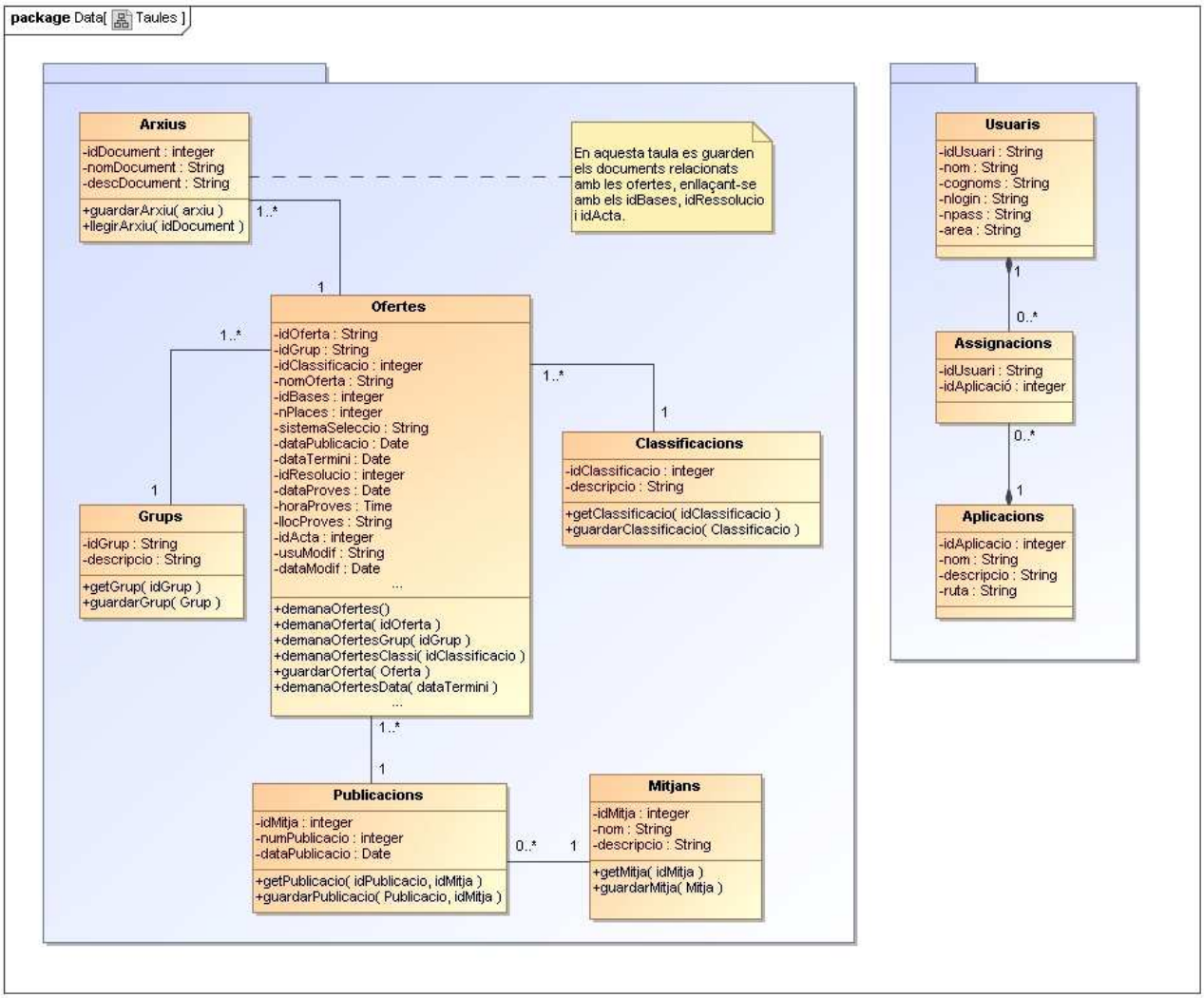
Aquestes classes constitueixen l'esquelet de l'aplicació, ja que en elles hi trobem totes les dades emmagatzemades.

Degut a facilitats en la seva relació i ús, n'he afegit algunes més, per exemple:

Classe Arxiu
Aquesta classe és la que utilitzaré en el emmagatzematge dels arxius pdf. Així, la oferta enlloc de contenir tots els pdf guardarà un identificador als diferents documents que estaran guardats en una classe d'aquest tipus.
- idDocument (*integer*)
- nomDocument (*string*): Nom de l'arxiu guardat
- descDocument (*string*): Descripció del document guardat.

Ara anem a veure el diagrama de classes amb les seves diferents relacions. Els he separat en dos grups de classes, el grup relacionat amb la gestió d'usuaris i aplicacions i el grup de l'aplicació d'ofertes d'ocupació:

Diagrama de classes



En el diagrama de classes i associacions, es reflecteixen pràcticament les estructures de la base de dades. Per tant, veiem que per garantir la normalització de la base de dades, s'han afegit les classes:

- **Assignacions**, que és la que s'encarregarà d'emmagatzemar la relació entre els usuaris i les diferents aplicacions que aquests puguin utilitzar (Tot i que recordem que de moment només en tindran una, més endavant aquest nombre s'ha d'ampliar).

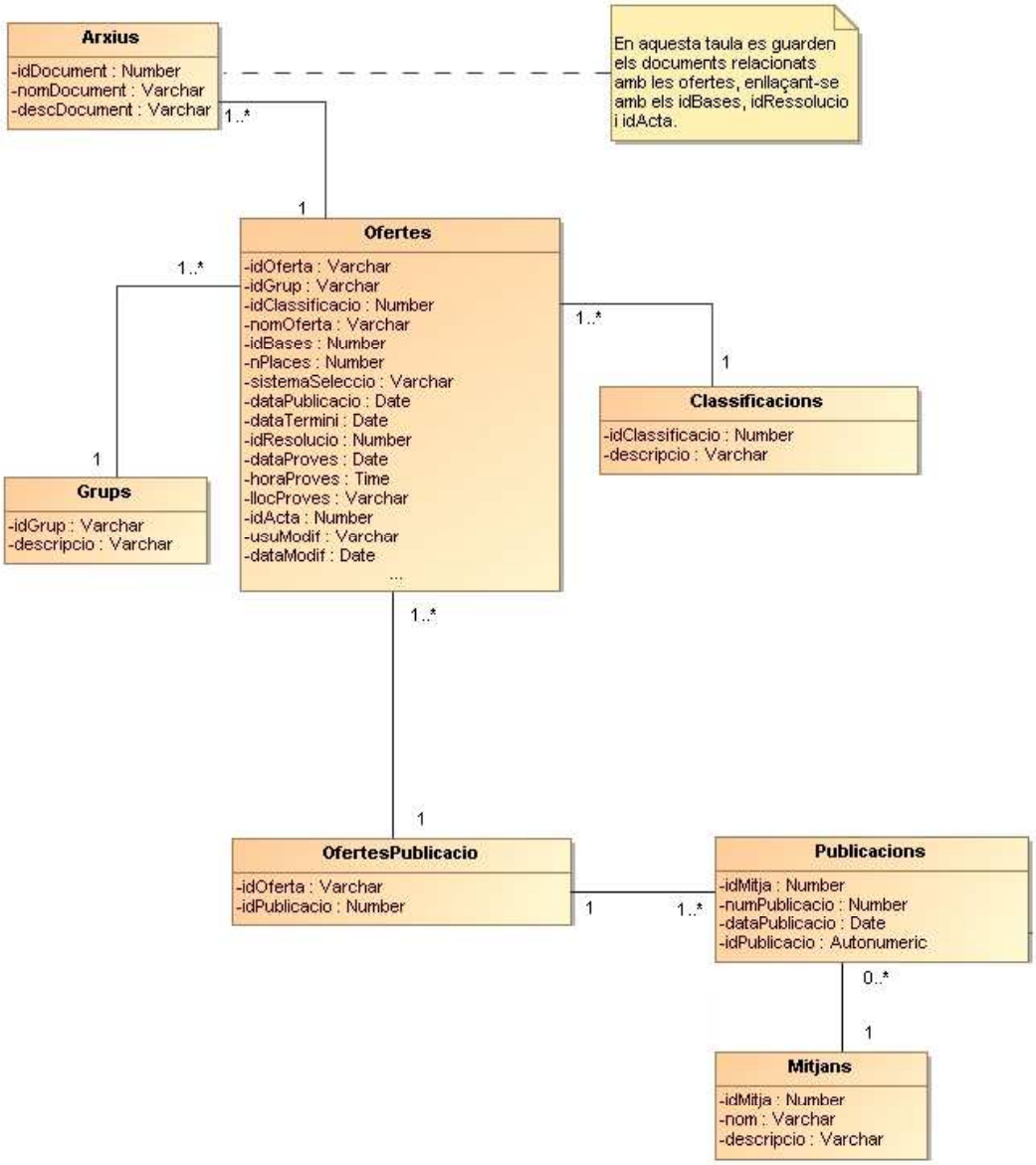
- **PubliOfertes**, és on enllaçarem les diferents ofertes i en quins mitjans han sortit publicades.

6.2 - Definició de la base de dades

Ara es definiran les diferents bases de dades. En primer lloc, fem un cop d'ull a les classes que tenim definides per veure que utilitzarem una taula per cada classe que hi ha, i els diferents atributs de la taula seran els diferents atributs de la classe.

Per tant, gràcies a les classes que hem trobat als casos d'ús, veiem les taules de la base de dades que crearem.

Taules de la aplicació



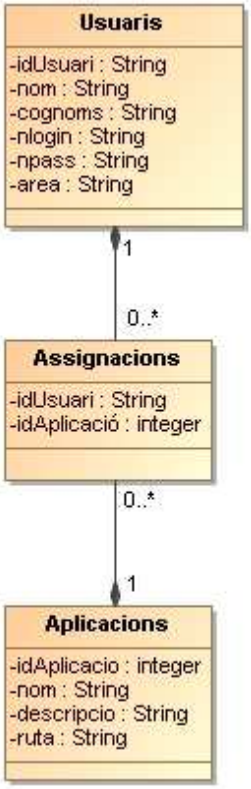
Es veu que una de les diferències amb el diagrama de classes és la existència de la taula intermitja

OfertesPublicacio.

Això és així perquè es pot donar el cas de que un mitja publiqui un número amb varies ofertes incloses, i que una oferta es publiqui en més d'un número de diferents mitjans.

Per això s'ha creat aquesta taula, incloent els camps necessaris per identificar la oferta i la publicació (per fer això s'ha creat un camp de publicacions auto numèric, ja que així ens assegurem que no hi hagi problemes amb que dos mitjans coincideixin amb el seu número de publicació.

Taula de la part del gestor d'aplicacions



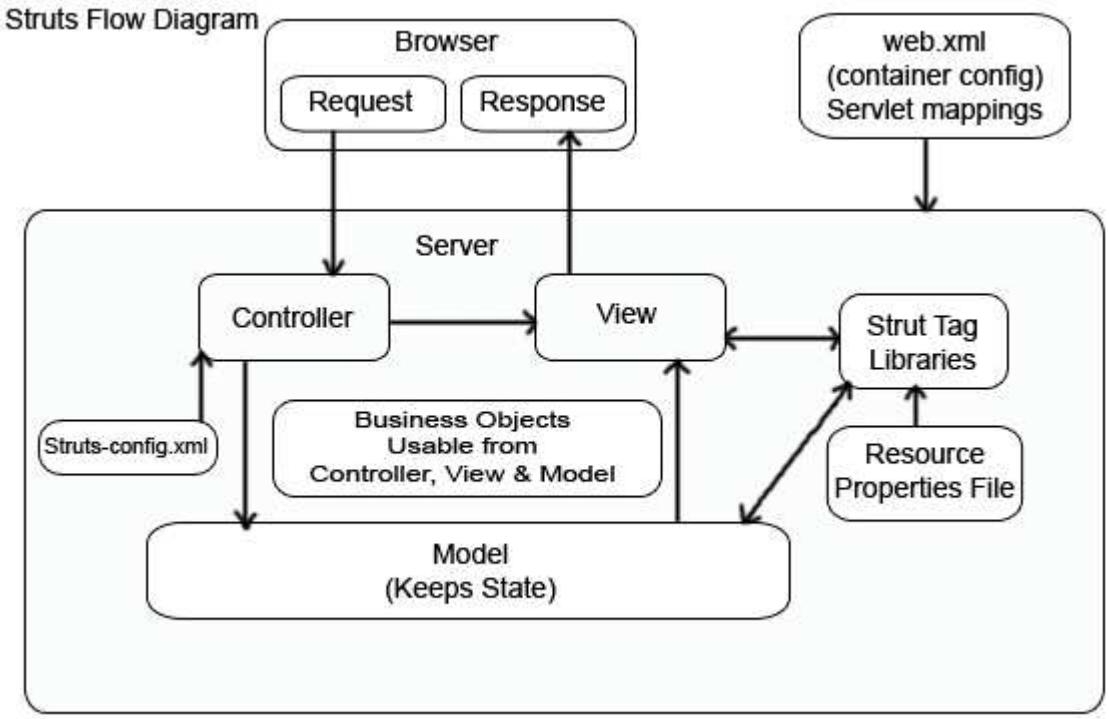
Capítol 7 – Disseny del funcionament del sistema

7.1 - Struts i el patró Model-Vista-Controlador

L'aplicació s'ha desenvolupat amb llenguatge JAVA i utilitzant el framework Struts.

Struts és una eina de suport per al desenvolupament d'aplicacions web sota el patró Model-Vista-Controlador (MVC). Aquest patró d'arquitectura del software es basa en separar les dades d'una aplicació, la interfície d'usuari i la lògica de control en tres components diferents.

En el nostre cas, tractant-se d'una aplicació web, la vista són les pàgines JSP, el model és el sistema de base de dades (MySQL) i les classes de lògica de negoci, i el controlador és el responsable de rebre els events de la entrada des de la vista.



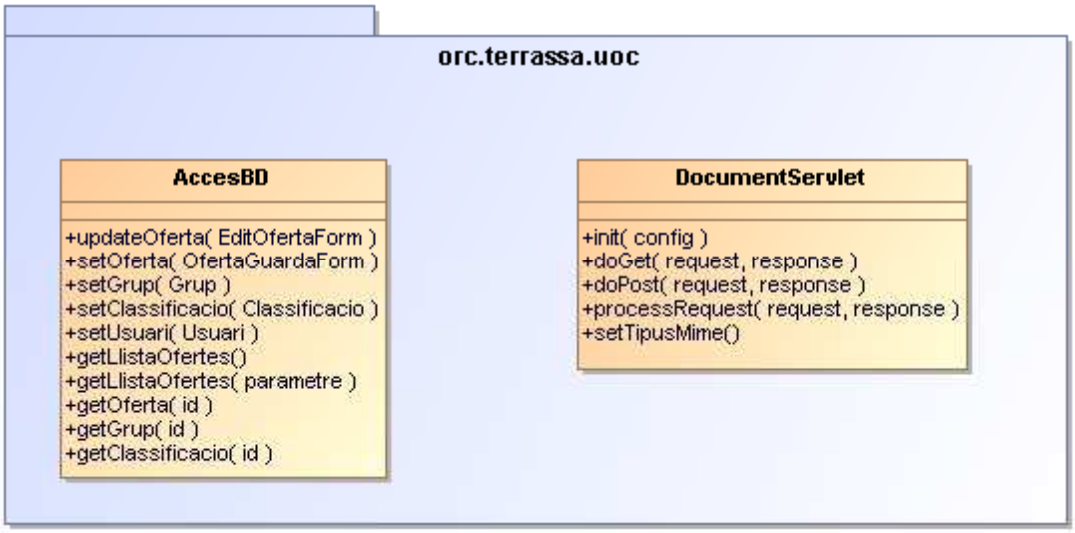
Tal com es veu en aquest diagrama, si identifiquem els seus elements:

•**web.xml**: S'encarrega de mapejar les peticions d'arxius al servlet d'acció corresponent.
•**Request (petició)**: Aquest és el segon pas realitzat pel contenidor després de revisat el *web.xml*. En ell, l'usuari envia un formulari i la petició és interceptada pel controlador.
•**Controller:** És el cor del contenidor. El controlador determina quina acció es requereix i envia la informació per ser processada per un bean d'acció.
•**Struts-config.xml:** Struts té un fitxer de configuració on emmagatzema els mapejos de les accions. És responsabilitat del controlador el validar el fitxer *struts-config.xml* per determinar quin mòdul s'ha de cridar degut a una petició d'acció.
•**Model**: El model és bàsicament una part de la lògica de negocique agafa la resposta de l'usuari i emmagatzema el resultat per la duració del procés. És un gran lloc on implementar el pre-processament de les dades rebudes d'una petició.
És possible reutilitzar el mateix model per diferentes peticions de pàgines. Struts ens proporciona les classes Action i ActionForm que poden ser exteses (extend) per crear els objectes del model.
•**Vista:** La vista al framework Struts són les pàgines jsp responsables de produir la sortida per l'usuari.
•**Struts tag libraries**: Són components de Struts que ens ajuden a integrar el framework dins de la lògica del projecte. Així, aquestes llibreries s'utilitzen dins de les pàgines jsp mitjançant crides de les seves etiquetes.
Això vol dir que les parts del controlador i el model no poden utilitzar la llibreria d'etiquetes de Struts, però poden utilitzar la seva llibreria de classes pel control dels processos.
•**Property File**: S'utilitza per guardar els missatges que un objecte o pàgina poden utilitzar. Els fitxers de propietats es poden utilitzar per guardar els títols i altres valors. Es poden tenir diferents arxius de propietats de diferents idiomes cadascun, i així facilitar el multilingüisme de l'aplicació.
•**Bussines objects (objectes de negoci)**: És on hi ha les normes de l'actual projecte. Són els mòduls que regulen l'activitat diària del lloc.
•**Response**: Sortida de l'objecte de vista jsp.

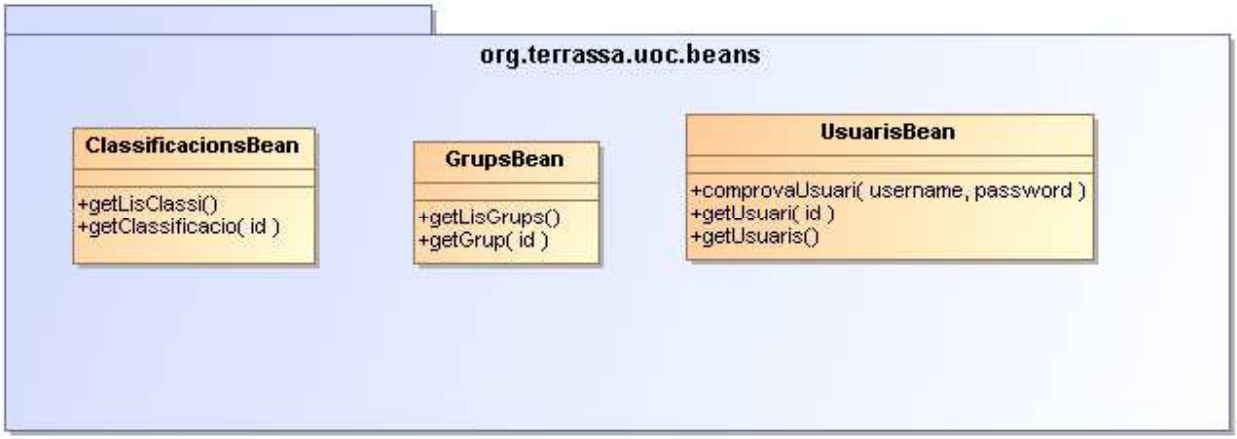
7.2 - Estructura del projecte

En el projecte els diferents tipus de servlets estan separats i agrupats en diferents paquets, així amb aquesta organització aconseguim que la nostra aplicació sigui més portable, reaprofitable i fàcil de mantenir.

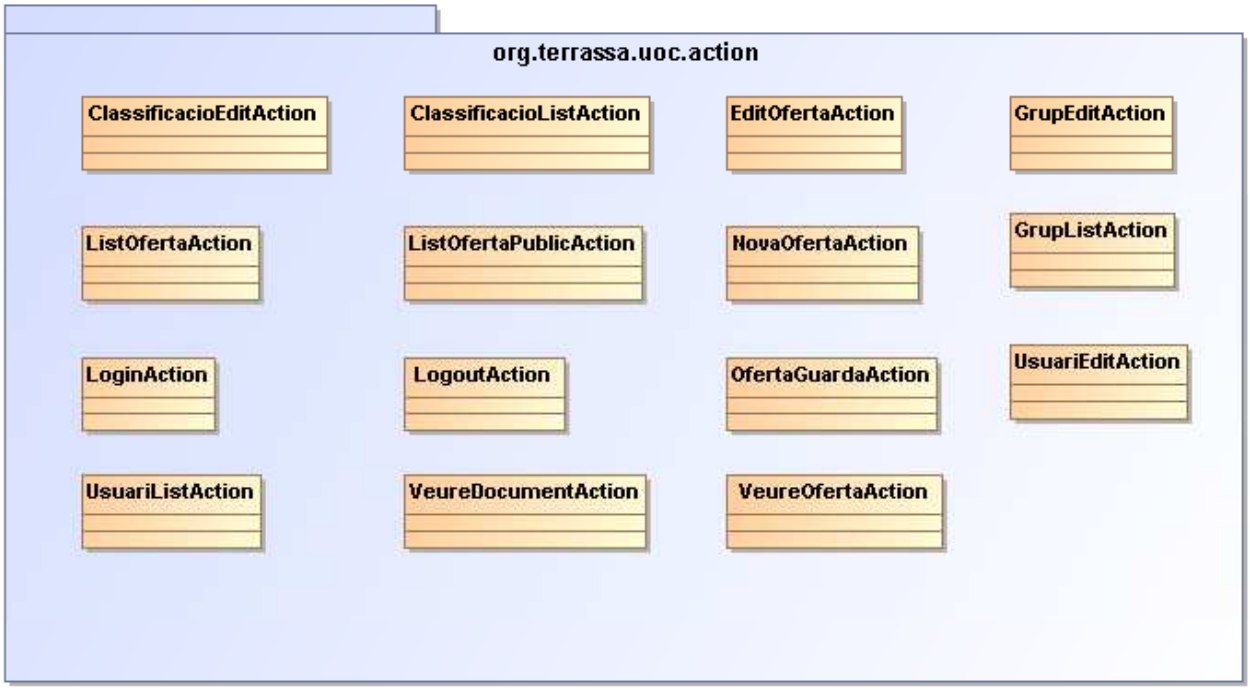
A l'arrel dels paquets del projecte hi tenim dues classes, la que conté les funcions que realitzen els canvis a la base de dades i la que ens visualitza els documents pujats a l'aplicació.



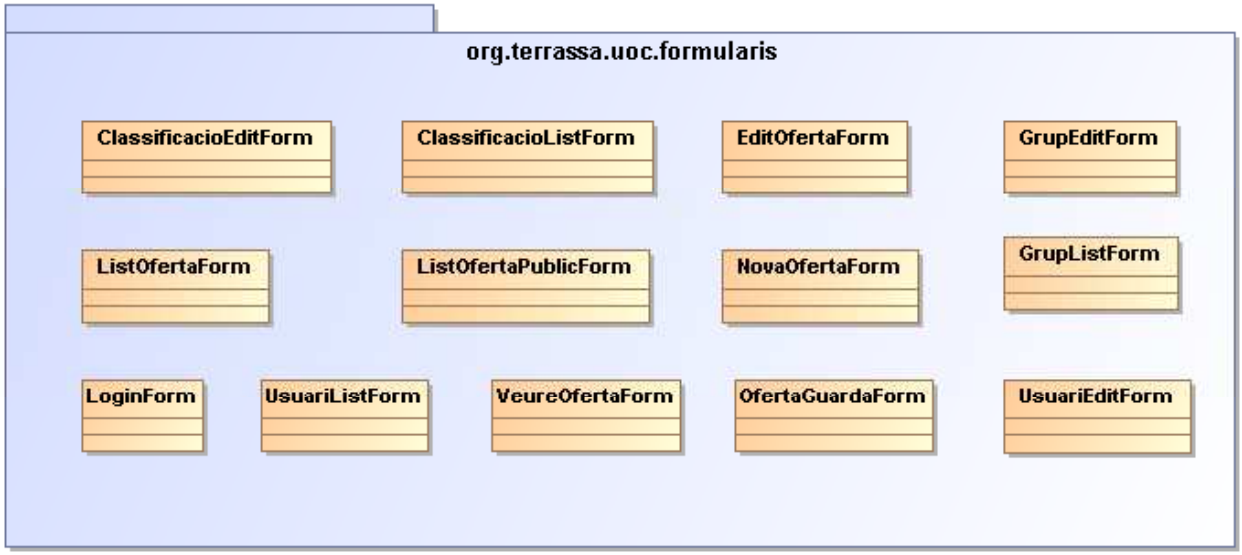
Dins del paquet org.terrassa.uoc.beans, tenim les classes que utilitzem per obtenir llistats i alguns objectes d'algun model definit.



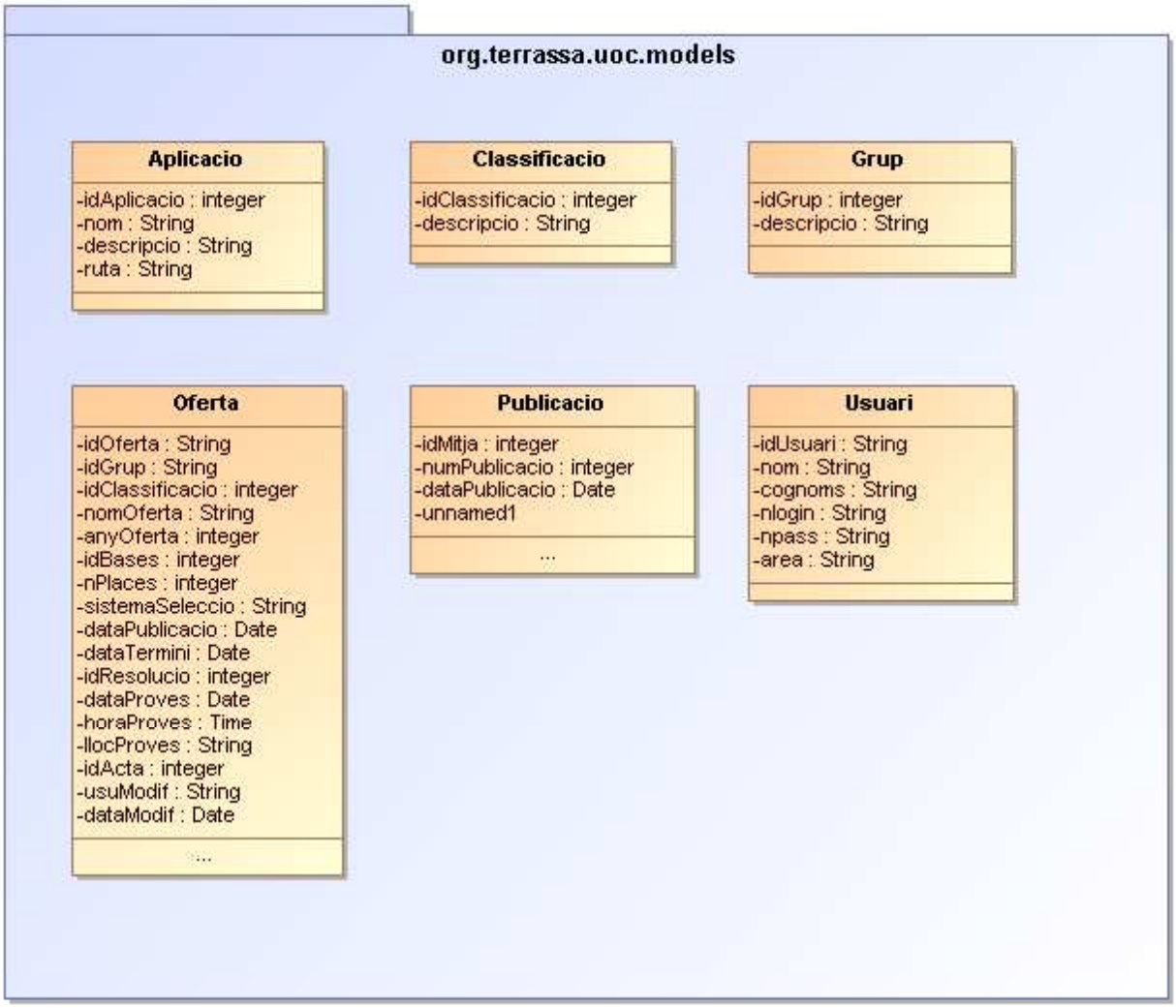
Dins del paquet org.terrassa.uoc.accions hi ha les diferents accions que es criden des del fitxer de configuració de Struts.



A org.terrassa.uoc.formularis tenim les classes de formularis que utilitzem tant a les accions com al publicar pàgines jsp.



I, finalment, dins del paquet org.terrassa.uoc.models tenim les classes creades com a objectes model. Aquests objectes ens faciliten la lectura dels elements de la base de dades i la seva assignació entre les diferents funcions de l'aplicació.



Capítol 8 – Funcionament del sistema

Aquest capítol és una guia per als usuaris de l'aplicació. És una breu explicació per als dos perfils d'usuaris de l'aplicació, el personal de l'Ajuntament de Terrassa que mantindran les dades de l'aplicació internament i els ciutadans navegadors d'internet que accediran a la web municipal per consultar les ofertes d'ocupació.

8.1 - Perfil del ciutadà

El ciutadà accedirà al portal de l'Ajuntament, des d'on se'l redirigirà cap al servidor de l'Administració Oberta (és el servidor que actualment és el servidor d'aplicacions java), o simplement des de la web municipal mitjançant un enllaç.

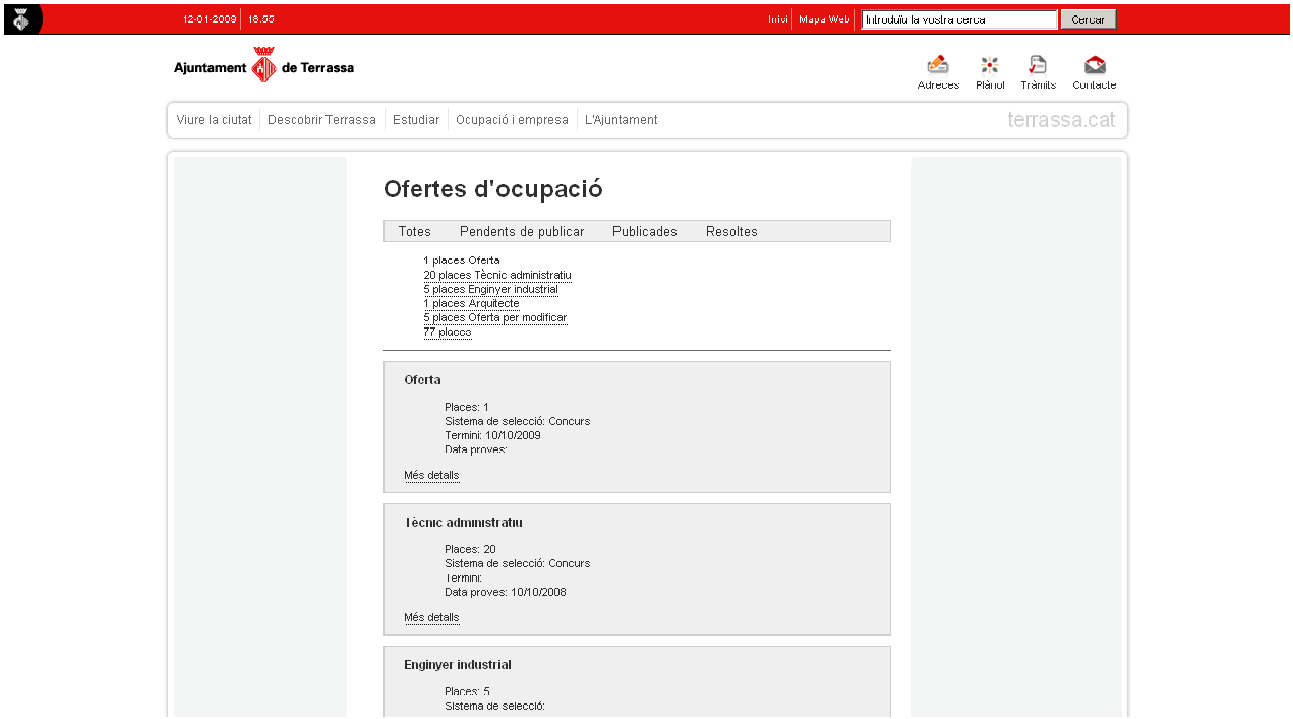
En aquesta pàgina el ciutadà veurà un llista de les ofertes d'ocupació que actualment hi ha publicades.

En primer lloc hi ha un selector gràcies al qual el ciutadà pot saber quines ofertes estan planificades però no publicades, en procés de publicació i adjudicades.

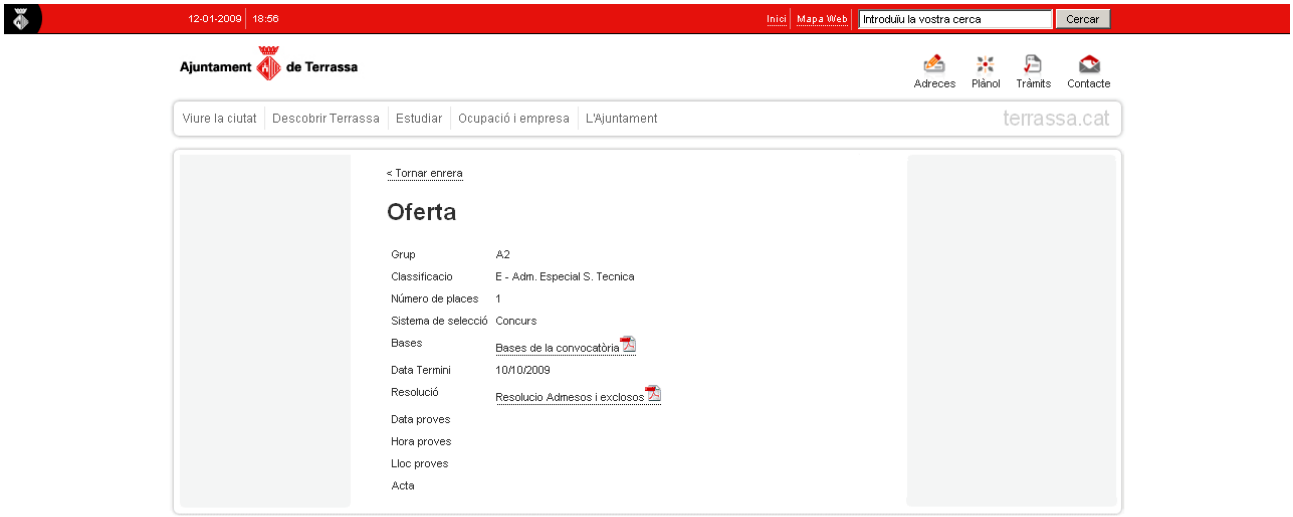
Tot seguit, hi ha un quadre amb el nom de totes les ofertes publicades i les places de cadascuna. Aquestes ofertes enllacen directament amb la oferta tal com es mostra al llistat dins de la mateixa pàgina. Si un cop estem a la oferta dins del lista i seleccionem el nom de la oferta o sobre l'enllaç "Més informació", el ciutadà anirà a la pàgina de detall de la oferta, on veurà totes les dades de la oferta, i es podrà descarregar els documents de les publicacions de les actes.

Aquesta part ha estat desenvolupada seguint el llibre d'estil de l'Ajuntament de Terrassa, i complint els criteris d'accessibilitat AA del W3C.

Llistat d'ofertes a internet



Detall d'una oferta a internet



8.2 - Perfil intern

Anem a veure ara com son les pantalles de l'aplicació. Aquesta part interna estarà instal·lada en un servidor d'accés intern, de manera que serà impossible accedir-hi des de fora de la xarxa municipal.

Per accedir al sistema, l'usuari s'identificarà en primer lloc, seguint l'enllaç pertinent. Un cop introduït el seu nom d'usuari i la seva clau (que en el cas de l'Ajuntament, serà el seu dni com a nom d'usuari i la clau que s'hagi configurat al donar d'alta a l'usuari).

Un cop identificat al sistema, l'usuari veurà a la columna esquerra de la pàgina les diferents opcions a les que pot accedir. Aquestes opcions són:

•Manteniment de Grups: Cada oferta d'ocupació pertany a un grup diferent depenent de la formació necessària per desenvolupar la plaça ofertada. Actualment la classificació és la que ve donada d'alta amb l'aplicació, però com que és sensible a canvis, s'ha habilitat aquest manteniment.

•A1: Enginyeries superiors
•A2: Enginyeries tècniques
•C1: Batxillerat
•C2: EGB
•APG: Agrupacions Professionals

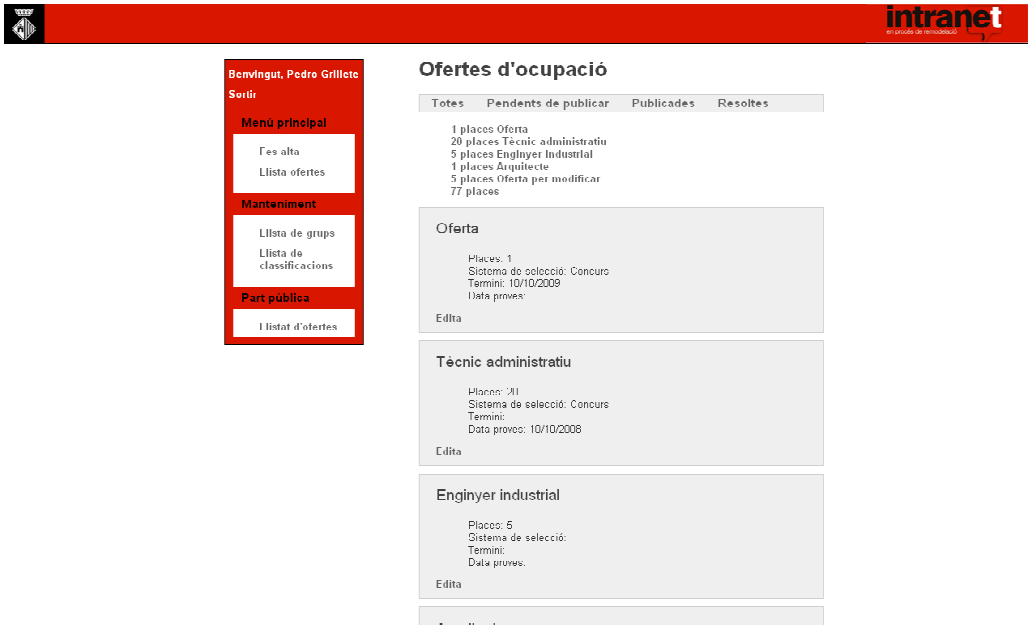
Pantalla de llistat de grups

<u>Manteniment de Classificacions</u>: Depenent de dins que quina escala s'englobin les ofertes, aquestes pertanyen a una classificació o a una altra. Al instal·lar l'aplicació, tindrem les classificacions per defecte instal·lades. Amb aquest enllaç entrem a la pàgina de manteniment de les classificacions:

- E – Adm. Especial S – Tècnica
- E – Adm. Especial S – Serv. Especials
- E – Adm. Especial S – Subalterna

•<u>Manteniment de Ofertes</u>: Aquí és on es mostrarà un breu llistat de totes les ofertes a la part superior de la pàgina, i el personal podrà anar a veure el detall de la oferta, seleccionant-ne el nom o seguint l'enllaç Més dades. Un cop fet això, podem editar les dades de la oferta i canviar les actes adjuntes.

Llistat d'ofertes a la intranet

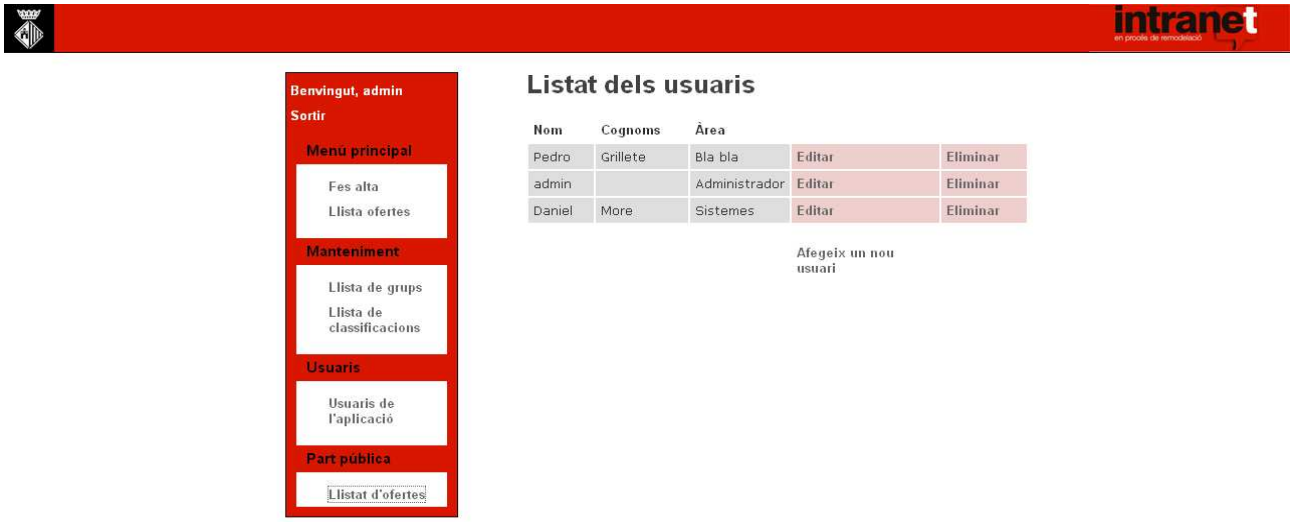


•Alta d'Oferta: Amb aquest enllaç entrarem a la pàgina d'alta de les ofertes. És una pantalla més simple que la d'edició, ja que quan es creen les ofertes únicament es coneixen aquestes dades per qüestió de plaços.

Pantalla d'edició d'una oferta

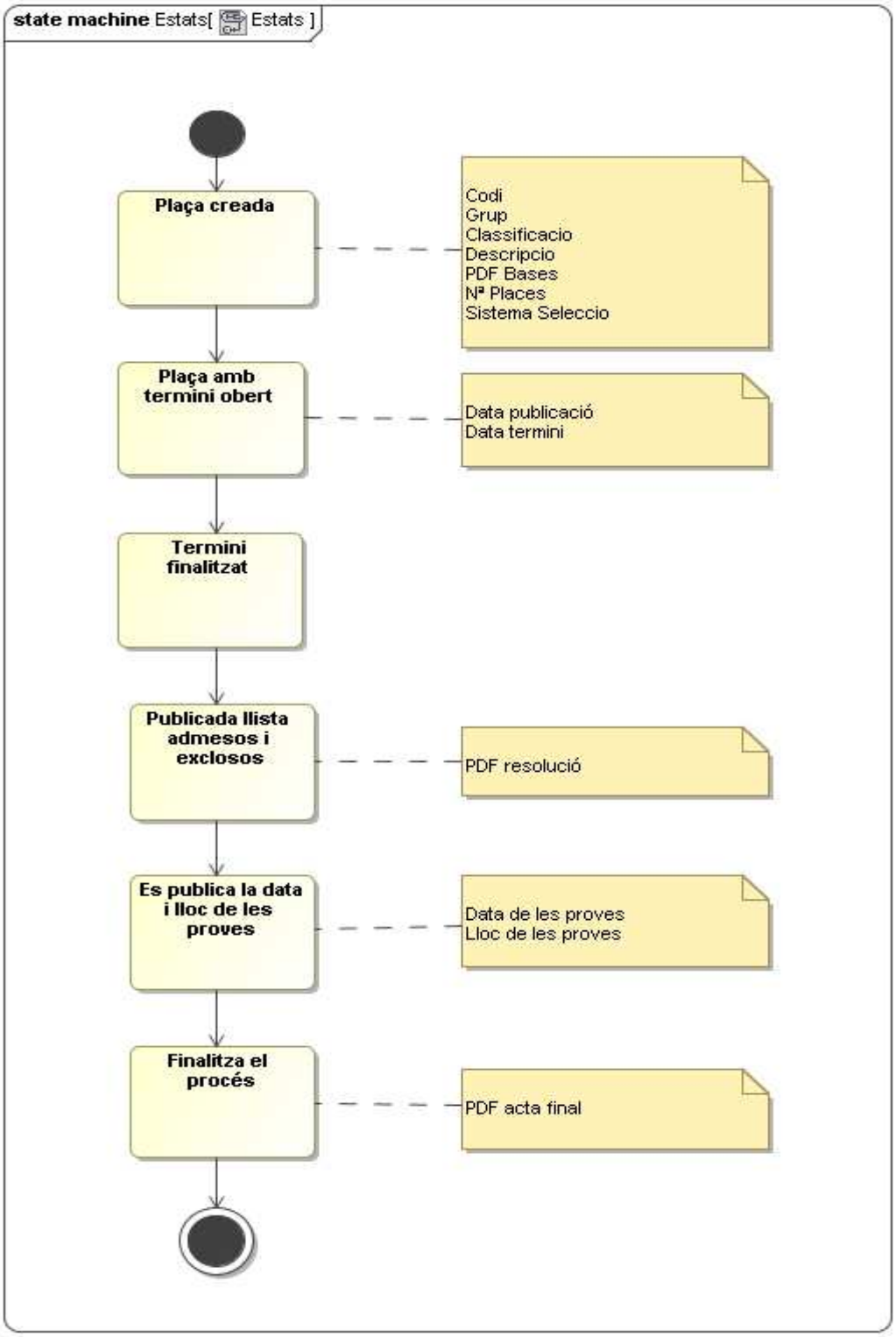
Si ens connectem al sistema amb l'usuari administrador, a més a més podrem accedir al manteniment dels usuaris, gràcies al qual l'administrador pot crear i editar els usuaris existents.

Llistat dels usuaris de l'aplicació



8.3 - Cicle de vida d'una oferta

Anem a veure quines són els diferents estats pels que passa una oferta. Tot seguit es mostra un gràfic en que es mostren els estats i quines són les dades que s'afegeixen a la oferta en cada estat. L'estat de la oferta depèn de quines dades estiguin plenes.



Capítol 9 - Conclusions i bibliografia

Per la realització d'aquesta aplicació he hagut d'aprendre com funciona el framework Struts, ja que anteriorment només havia programat anteriorment alguna aplicació amb java.

M'ha semblat una manera de treballar força interessant, ja que utilitzar el model model-Vista-Controlador facilita separar el desenvolupament de les diferents operacions i fins i tot la exportació d'algunes parts de la nostra aplicació.

La realització d'un producte que té uns usuaris reals que l'utilitzaran ha suposat un bon al·licient, ja que he treballat amb necessitats reals mitjançant entrevistes amb els futurs usuaris de l'aplicació. Aquest desenvolupament ha suposat la primera pedra d'una nova aplicació que aniré desenvolupant a mida que noves necessitats vagin apareixent.

Gràcies a aquesta aplicació, s'ha passat de mantenir unes dades mitjançat la edició de pàgines html a mantenir les dades en una base de dades, el que ens proporciona poder treballar amb aquestes per crear noves funcionalitats.

Vull destacar, en el meu procediment d'aprenentatge, l'ajut que m'ha suposat consultar les següents webs i comunitats:

- Notes d'instal·lació de java: http://java.sun.com/j2se/1.5.0/install-windows.html
- Lalíluna, java tutorials and development: http://www.lalíluna.de/index.html
- El blog de leejeok: http://leejeok.wordpress.com/2008/05/25/struts-build-html-options-tag-elements-from-mysql/
- El desarrollo del framework orientado al objeto: http://www.acm.org/crossroads/espanol/xrds7-4/frameworks.html
- The Server Side: http://www.theserverside.com

També he adquirit un llibre que m'ha servit per iniciarme pas a pas en aquesta tecnologia i que espero que em pugui seguir ajudant a mida que segueixi desenvolupant amb Struts.

- **Apache Struts**, per Arnold Dorian. Editorial Anaya Multimedia, ISBN 978-84-415-2216-9