

Universitat Oberta de Catalunya



Treball Final de Grau – Java EE

Grau d'Enginyeria Informàtica (Enginyeria del Programari)

Consultor: Albert Grau Perisé

The Job Board

Borsa de treball on-line, distribuïda i multi-dispositiu

Memòria

Josep Itarte Aguiló

Gener 2014

Índex

1. Introducció.....	6
2. Descripció del projecte.....	7
2. 1 Descripció general.....	7
2. 2 Objectius.....	8
2. 2. 1 Principals	8
2. 2. 2 Secundaris	8
2. 3 Funcionalitats.....	9
3. Metodologia.....	12
3. 1 Gestió del projecte.....	12
3. 2 Fases de desenvolupament.....	13
4. Anàlisi de requisits.....	14
4. 1 Característiques generals.....	14
4. 2 Requisits funcionals.....	14
4. 3 Requisits no funcionals.....	17
5. Especificació.....	18
5. 1 Actors del sistema.....	18
5. 2 Diagrama de casos d'ús.....	19
5. 3 Especificació de casos d'ús.....	20
5. 3. 1 Casos d'ús: Gestió d'usuaris.....	20
5. 3. 2 Casos d'ús: Gestió d'ofertes.....	23
5. 3. 3 Casos d'ús: Gestió d'accés.....	29
5. 3. 4 Casos d'ús: Estadístiques.....	30
5. 4 Model conceptual.....	31
5. 4. 1 Model de dades no normalitzat.....	32
5. 4. 2 Model de dades normalitzat.....	33
6. Disseny.....	34
6. 1 Arquitectura de l'aplicació.....	34
6. 1. 1 Patró Model-Vista-Controlador.....	34
6. 1. 2 Cicle de vida d'una petició HTTP.....	35

6. 1. 3 Estructura de directoris.....	37
6. 2 Diagrama de seqüència.....	38
6. 2. 1 Diagrama d'una petició HTTP.....	38
6. 3 Configuració de play framework.....	40
6. 3. 1 Mode desenvolupament.....	42
6. 3. 2 Mode producció.....	43
7. Implementació.....	45
7. 1 Programari utilitzat	45
7. 2 Capa de persistència.....	46
7. 3 Capa de negoci.....	48
7. 4 Capa de presentació	49
7. 5 Verificació.....	50
8. Planificació.....	51
8. 1 Planificació inicial	51
8. 2 Planificació final.....	53
8. 3 Avaluació de costos.....	54
9 Avaluació.....	55
9. 1 Objectius aconseguits.....	55
9. 2 Treball futur.....	56
9. 3 Codi font.....	57
9. 4 Aplicació desplegada.....	57
10. Conclusions.....	58
11. Glossari de termes.....	60
12. Bibliografia.....	62
12. 1 Documentació oficial.....	62
12. 2 Articles en línia.....	63
13. Manual d'usuari.....	65

Índex de figures

Figura 1: Esquema dels actors del sistema.....	18
Figura 2: Diagrama de casos d'ús de l'aplicació.....	19
Figura 3: Model conceptual de dades.....	32
Figura 4: Model conceptual de dades normalitzat.....	33
Figura 5: Esquema simplificat de Play Framework.....	35
Figura 6: Cicle de vida de play framework.....	36
Figura 7: Diagrama de seqüència de petició HTTP Play Framework.....	39
Figura 8: Diagrama UML de la capa de persistència.....	47
Figura 9: Diagrama del controladors de l'aplicació.....	48
Figura 10: Fixers de les vistes.....	49
Figura 11: Planificació inicial de tasques amb les jornades.....	52
Figura 12: Diagrama de Gantt.....	53
Figura 13: Pàgina principal de l'aplicació (navegador).....	65
Figura 14: Opcions de nou compte d'usuari.....	65
Figura 15: Opcions de canvi d'idioma.....	66
Figura 16: Pàgina de creació d'usuari de tipus Candidat.....	66
Figura 17: Pàgina de creació d'usuari tipus Empresa.....	67
Figura 18: Pàgina d'accés a l'aplicació.....	67
Figura 19: Pàgina principal de l'àrea Candidat.....	68
Figura 20: Opció de cercar noves ofertes.....	68
Figura 21: Actualització del compte d'usuari.....	69
Figura 22: Pàgina principal d'usuaris de tipus Empresa.....	69
Figura 23: Opció per veure usuaris candidats inscrits.....	70
Figura 24: Llistat d'inscrits.....	70
Figura 25: Pàgina per publicar noves ofertes.....	71
Figura 26: Actualització de perfil d'usuari empresa.....	71
Figura 27: Pàgina principal de l'usuari de tipus Administrador.....	72
Figura 28: Administrador d'usuaris candidats del sistema.....	72
Figura 29: Administració d'empreses del sistema.....	73
Figura 30: Gestió de les ofertes de treball del sistema.....	73
Figura 31: Actualització de dades d'Administrador.....	74

Figura 32: Pàgina sobre les tecnologies utilitzades.....	74
Figura 33: Pàgina amb informació de contacte.....	75
Figura 34: Aplicació adaptada a tauleta.....	76

1. Introducció

Aquest document és la memòria descriptiva del Treball Final de Grau de títol *The Job Board, borsa de treball on-line, distribuïda i multi-dispositiu*, desenvolupat per Josep Itarte que es correspon amb el *Grau d'Enginyeria Informàtica* de la Universitat Oberta de Catalunya.

A grans trets, el propòsit d'aquest projecte és la creació i la posada en funcionament d'una borsa de treball on-line on empresaris puguin trobar treballadors i viceversa, amb l'objectiu de fomentar l'activitat l'aboral.

La memòria descriu de manera global tots els aspectes que s'han hagut de tenir en compte per assolir els objectius del projecte. S'explica quin és el context del qual es parteix, quines són les necessitats que han motivat la creació del projecte i quines són les tasques que s'han realitzat al llarg de tot el procés de treball.

2. Descripció del projecte

2.1 Descripció general

En el moment social actual, la situació empresarial, i especialment contractual, és molt problemàtica. Qualsevol ajuda per millorar el problema de la desocupació ha de ser benvinguda, i possiblement, una nova xarxa de comunicació entre empreses i futurs treballadors pot ser interessant. Amb aquesta idea neix el Treball de Final de Grau.

El TFG té com a principal objectiu la creació d'una borsa de treball on-line on diferents perfils d'usuari puguin gestionar ofertes de treball i establir contacte. Les empreses podran publicar-hi ofertes de treball per cercar treballadors, i els usuaris que vulguin trobar feina podran cercar ofertes també per mitjà de l'aplicació.

Actualment ja existeixen diverses borses de treball on-line en funcionament, algunes amb molt èxit i conegudes. Però existeixen diversos trets diferencials entre el TFG i aquestes eines, que es podrien resumir en el següent: no serà de pagament i s'implementarà amb tecnologies noves. Tot seguit es donen més detalls:

Els usuaris de tipus empresa que publiquin les ofertes caldrà que omplin un formulari amb la informació necessària, com per exemple, nom de l'empresa, requisits que hauria de tenir l'aspirant, les feines que hauria de dur a terme, tipus de contracte, dades de contacte, etc. Per altra banda, els usuaris que desitgin trobar feina hauran de fer-ho usant un formulari diferent amb el qual es basaran els criteris de cerca.

Hi haurà un perfil d'usuari administrador que tindrà privilegis per modificar, eliminar o actualitzar les dades.

Tots els perfils d'usuari caldrà que disposin d'un sistema de seguretat per accedir a les seves pàgines personals (com per exemple un sistema de login).

Els usuaris que publiquen ofertes de treball podran compartir aquesta informació tot incloent-la de manera senzilla en les principals xarxes socials.

2. 2 Objectius

2. 2. 1 Principals

- Creació d'una aplicació web que permeti la comunicació entre empreses i treballadors per fomentar la contractació. Les empreses publiquen ofertes de feina i els candidats s'hi poden inscriure.
- Una aplicació web que pugui ser accessible des de qualsevol dispositiu connectat a la Internet. Ha de ser per tant poc complexa i molt intuïtiva.
- Desplegar l'aplicació al núvol fent-la pública per a qualsevol usuari i també oferir la possibilitat d'integrar-la com a borsa de treball pròpia d'una empresa.

2. 2. 2 Secundaris

- Crear un projecte, que finalitzarà amb un producte acabat, tot seguint i utilitzant els continguts apresos al llarg de les assignatures cursades al Grau d'Enginyeria Informàtica. Concretament, usant com a pedra angular les tecnologies Java.
- Ampliar els coneixements apresos sobre desenvolupament d'aplicacions web sobretot usant tecnologies noves i punteres.
 - Programació avançada: presa de contacte amb Play! (playframework) com a marc de treball per al desenvolupament web. Aprendre a usar-lo i a codificar l'aplicació web usant aquesta metodologia.
 - Llenguatge de programació Scala: aprendre a programar amb un nou llenguatge de programació funcional (Scala) per a la capa de presentació.
 - Connexió amb Bases de dades: l'ús d'un framework ORM aportarà un nou coneixement sobre l'abstracció dels models de dades.
 - Interfícies gràfiques: millorar i ampliar els coneixements sobre HTML5 i CSS3 per garantir que l'aplicació web sigui usable en diversos dispositius.
 - Seguretat: ampliació dels coneixements en l'àmbit de la seguretat web tot implementant un sistema de seguretat login.

2. 3 Funcionalitats

En aquest apartat s'expliquen amb més detall les funcionalitats que s'implementaran i també es classifiquen segons la seva prioritat. Per esquematitzar tot el seguit de funcionalitats, s'ha fet una primera separació tenint en compte la temàtica de la funcionalitat, i posteriorment una prioritització de les mateixes.

i. Gestió d'usuaris

- i. *Alta d'usuaris*: aquesta funcionalitat ha de permetre als nous usuaris poder registrar-se a l'aplicació omplint amb algunes dades un formulari. El registre permet diferenciar entre tres perfils diferents: el perfil candidat que correspondria als usuaris que volguessin inscriure's en una oferta de feina, el perfil empresa que correspondria als empresaris que volguessin publicar ofertes, i el perfil administrador, encarregat d'assegurar el bon estat de les dades del sistema.
- ii. *Baixa d'usuaris*: en aquesta simplement es permet donar de baixa a usuaris que estiguin registrats en el sistema.
- iii. *Modificació de les dades d'usuari*: cada usuari podrà accedir al seu compte personal per actualitzar les seves dades personals.
- iv. *Incorporació de documents*: aquesta funcionalitat ha de permetre als usuaris registrats (de tipus candidat) que puguin aportar documentació (CV, certificats, etc.) al seu compte personal que pugui ser consultat per les empreses. Només podran tenir accés les empreses a les quals el candidat hagi aplicat a una oferta. Ha de garantir el compliment de la LOPD.

ii. Gestió d'ofertes de treball

- i. *Alta d'ofertes de treball*: els perfils de tipus empresa han de poder publicar ofertes de treball tot omplint amb algunes dades requerides els camps d'un formulari. Un dels camps més importants serà la tipologia contractual, si és una feina de tipus freelance, contracte laboral indefinit, etc.
- i. *Millora, caducitat de les ofertes de treball*: les ofertes publicades per les empreses poden disposar d'un control de caducitat que caldrà especificar en

el moment de publicació de les mateixes. Aquest control permetrà descartar automàticament les ofertes més antigues.

- ii. *Baixa d'ofertes*: també és necessari que aquestes ofertes puguin ser donades de baixa o marcades com a processos finalitzats.
- iii. *Modificació de les dades de les ofertes*: de la mateixa manera, els usuaris de tipus empresa també podran actualitzar algunes dades de les mateixes ofertes, per fer-hi aclariments o corregir errors.
- iv. *Llistat d'ofertes de feina publicades*: els usuaris de tipus empresa podran consultar les ofertes que han estat publicades i conèixer en quin estat estan (vigents, caducades, etc.) i també saber la quantitat d'inscrits.
- v. *Llistat d'ofertes inscrites*: els usuaris de tipus candidat podran veure el llistat d'ofertes a les quals s'han inscrit com a candidats, com també veure la quantitat total de candidats competidors.
- vi. *Cercador d'ofertes*: els usuaris de tipus candidat també podran fer una cerca exhaustiva d'ofertes de treball publicades tot utilitzant un formulari amb criteris de cerca. D'aquesta manera podran buscar de manera més acurada les ofertes que més els hi interessin.
- vii. *Inscripció a una oferta com a candidat*: els usuaris de tipus candidat podran inscriure's a les ofertes que creguin oportunes com a aspirants. En aquest moment l'empresa podrà accedir a la informació del candidat per, si és el cas, posar-se en contacte amb el mateix.
 - i. Millora, *comunicació candidat a empresa*: enviar correu electrònic a l'empresa quan un candidat s'inscriu a una oferta de la mateixa.
- viii. *Selecció del candidat*: en el moment en que una empresa ha seleccionat un candidat com a escollit podrà notificar a l'usuari que ha estat seleccionat.
 - i. Millora, *comunicació empresa a candidat*: enviar també en aquest cas un correu al candidat notificant-lo de que ha estat seleccionat juntament amb les instruccions sobre quins procediments ha de seguir.

iii. Seguretat

- i. *Login*: cal que existeixi un sistema de seguretat login per garantir l'accés segur a l'aplicació dels usuaris registrats i també la visualització segura de les seves pàgines personals

iv. Administració

- i. *Administració d'usuaris*: els usuaris de tipus administrador tindran accés al llistat d'usuaris de l'aplicació i tindran l'opció de donar-los de baixa o actualitzar-los.
- ii. *Administració d'ofertes de treball*: aquests tipus d'usuari també podran donar de baixa o modificar dades de les ofertes de treball publicades.

v. Estadístiques

- i. *Llistats estadístics*: processos estadístics que puguin ser d'interès als usuaris segons diversos paràmetres.

3. Metodologia

En els projectes de desenvolupament de programari, tenint en compte la seva complexitat i el grau d'incertesa que hi ha a l'hora de resoldre les fases de treball, és molt necessari definir una metodologia per a gestió del projecte concreta, de tal manera es pot agilitzar i formalitzar tant la gestió pròpia del treball com la del seu desenvolupament.

En aquest apartat es mostra quina és la metodologia que es seguirà per a la gestió del projecte, i a part, també s'introduiran les fases o etapes amb les que es planificarà el projecte.

3. 1 Gestió del projecte

El desenvolupament de programari, i especialment en la construcció d'aplicacions web les quals integren diverses eines i tecnologies, pot comportar complexitat a l'hora de definir amb claredat el temps a dedicar a cada tasca, ja que aquesta disciplina presenta un alt grau d'incertesa en totes les seves fases.

Per coordinar i gestionar totes les tasques que caldrà realitzar al llarg del projecte es farà servir una *metodologia clàssica o en cascada*, molt utilitzada en projectes de programari.

En l'apartat de planificació es pot veure un diagrama de Gantt on s'hi veuen representades totes les etapes del projecte juntament amb les seves sub-tàsques. En el diagrama també s'hi representa gràficament el temps que caldrà dedicar-hi, com també quines són les tasques que depenen de les altres.

A l'inici del projecte es fa una aproximació de les estimacions, però al llarg del procés es fa una revisió acurada de la planificació amb el coordinador per tal d'ajustar-la amb les circumstàncies de cada moment. A continuació es mostren les fases més importants del projecte.

3. 2 Fases de desenvolupament

Les fases que formen una metodologia en cascada són les següents, i cadascuna d'elles està condicionada per la seva fase anterior. Només es pot començar una fase si l'anterior està finalitzada.

Fase	Descripció
1. Anàlisi de requisits	<i>Definició clara del que cal desenvolupar i fins a on cal arribar. És necessari que els requisits cobreixin totes les necessitats inicials</i>
2. Especificació	<i>A partir dels requisits ja acceptats, cal fer un anàlisi del sistema on s'expressen les funcionalitats de la manera més detallada possible.</i>
3. Disseny	<i>En aquest punt es refinarà l'especificació anterior i es dissenyarà també l'estructura de dades.</i>
4. Implementació	<i>Moment per codificar l'aplicació sempre seguint el model dissenyat anteriorment.</i>
5. Verificació	<i>Executar un seguit de bateries de prova i comprovar que les necessitats exposades inicialment són cobertes</i>

4. Anàlisi de requisits

Aquest capítol es basa en la captura i l'anàlisi en profunditat dels requeriments del TFG. L'elaboració d'un bon anàlisi de requisits és clau ja que, qualsevol serrell mal definit, pot impactar de manera greu en qualsevol de les següents fases del projecte. Aquest capítol, doncs, és la pedra angular del cicle de desenvolupament. D'aquesta manera caldrà deixar ben clars els límits i les particularitats del que seran les noves funcionalitats. Una bona definició de requeriments estalviarà que, en tasques de desenvolupament, s'hagi de modificar feina ja feta.

Pel que fa a l'abast del TFG, es pot prendre com a negociació amb els stakeholders¹ del projecte els principals intercanvis d'opinions amb el tutor del projecte. En els moments inicials del projecte, aquest va detallar alguns punts necessaris que s'han considerat com a requisits funcionals i no funcionals.

4. 1 Característiques generals

Com ja s'ha apuntat anteriorment, l'aplicació a desenvolupar consistirà en una borsa de treball per tal de posar en contacte candidats i empreses. Les empreses podran publicar ofertes de feina per a que altres usuaris puguin inscriure's si els interessa. A banda d'això, també serà necessària una gestió d'usuaris i segons diferents tipologies (Candidat, Empresa o Administrador). A banda d'aquesta funcionalitat, també hi ha l'opció d'implementar un seguit d'estadístiques o llistats que puguin ser d'interès per als usuaris.

4. 2 Requisits funcionals

En aquest punt es descriuen els requisits que tenen a veure directament amb les funcionalitats que ha de tenir el projecte. En aquest cas es divideixen en diversos grups per a modular els requisits segons la seva tipologia.

¹ També es poden anomenar així als interessats o involucrats en un projecte. (Clients, proveïdors, etc.).

- **Gestió d'usuaris**

- *Alta d'usuaris*

- Registrar nous usuaris a partir d'un formulari
- Han de poder ser de tipus Candidat, Empresa o Administrador

- *Baixa d'usuaris*

- Donar de baixa usuaris del sistema
- Només es poden donar de baixa els usuaris tipus Candidat i Empresa

- *Modificació d'usuaris*

- Permet veure i modificar les dades del propi usuari
- Només poden modificar les seves dades els usuaris tipus Empresa i Candidat

- *Incorporació de documents*

- Permet adjuntar documents per complimentar la informació personal
- Limitat a usuaris Candidat o Empresa

- **Gestió d'ofertes**

- *Alta d'ofertes de treball*

- Els usuaris de tipus Empresa han de poder crear i publicar noves ofertes de treball a partir d'un formulari

- *Baixa d'ofertes de treball*

- Els usuaris de tipus Empresa han de poder donar de baixa ofertes de treball que hagin publicat.

- *Modificació d'ofertes de treball*

- Els usuaris de tipus Empresa han de poder veure i poder modificar les dades de les ofertes de treball que hagin publicat.

- *Llistat d'ofertes publicades*

- Els usuaris de tipus Empresa han de poder veure un llistat amb la totalitat de

les ofertes que han publicat al sistema.

– *Llistat d'ofertes inscrites*

- Els usuaris de tipus Candidat han de poder veure un llistat amb totes les ofertes en les quals s'hi han inscrit

– *Cercar ofertes*

- Els usuaris de tipus Candidat han de poder cercar ofertes a partir d'un formulari de cerca.
- A partir dels resultats d'aquest llistat els usuaris poden accedir al detall de les ofertes de treball.

– *Inscripció a una oferta*

- Els usuaris de tipus Candidat han de poder inscriure's en les ofertes de feina que creguin interessants.

– *Selecció de candidat*

- Els usuaris de tipus Empresa han de poder seleccionar d'entre els diferents candidats inscrits en una oferta el candidats que més s'hi adaptin

• **Sistema d'accés**

– *Login/Logout*

- El sistema ha de poder garantir l'accés amb seguretat a l'aplicació i garantir que els usuaris només poden veure les seves àrees privades.
- Els usuaris han de poder fer logout a l'aplicació.

• **Estadístiques**

– *Sistema d'estadístiques*

- Els usuaris de tipus Administrador han de poder accedir a una àrea on s'hi poden executar estadístiques per analitzar l'estat de l'aplicació.

4. 3 Requisits no funcionals

Aquests requisits són els que el projecte haurà de complir per garantir els aspectes que no tenen a veure directament amb la funcionalitat. Són necessaris ja que, tot i aconseguir una bona implementació d'una funcionalitat, pot quedar inutilitzable si el sistema no és usable, insegur o totalment ineficient. Tot seguit es detallen els més importants

- **Eficiència**

És necessari que la resposta per part del sistema sigui ràpida i fiable. El fet d'implementar una nova aplicació ha de fer que aquesta sigui el més ràpida possible, ja que d'aquesta manera la satisfacció dels usuaris serà major i més ràpida.

- **Usabilitat i interfície d'usuari**

És crucial que l'aplicació sigui clara, intuïtiva i fàcil d'utilitzar. Cal que el temps d'aprenentatge per part dels usuaris sigui el més curt possible.

- **Seguretat**

És necessari disposar d'un control d'accés a les dades dels usuaris. Tant els candidats com les empreses només han de poder veure les seves pròpies dades i les dades de tercers per contacte. És a dir, els candidats poden veure les dades de les empreses a les quals s'hi han inscrit a alguna oferta, i viceversa. Caldrà complir les lleis de protecció de dades LOPD.

- **Diversos idiomes**

És molt important que l'aplicació pugui ser visualitzada en diversos idiomes. D'aquesta manera s'obté un major volum de possibles usuaris que la utilitzaran.

- **Escalabilitat**

L'ampliació de l'aplicació també és molt important. Si existeix un volum d'usuaris molt elevat és necessari que aquesta es pugui ampliar i escalar de manera senzilla i que no sigui necessari tocar el codi de l'aplicació.

5. Especificació

Aquest apartat definirà el comportament del sistema a implementar. Per a fer-ho, caldrà en primer lloc tenir en compte els requisits obtinguts en l'apartat anterior. Amb aquests requeriments caldrà aleshores decidir quins són els actors del sistema, quins casos d'ús poden realitzar i en quins escenaris.

5. 1 Actors del sistema

Com s'ha definit en els requisits, l'aplicació disposarà d'un sistema de seguretat per tal de protegir les dades privades d'un usuari de la resta. Per fer-ho caldrà dotar de privilegis als usuaris. Com a conseqüència, hi haurà dos grans tipus d'usuaris, els que no estan fet login a l'aplicació i els que sí. A part d'això, els usuaris que han aportat les seves credencials al sistema i per tant han fet login, es podran dividir segons el seu perfil: Candidat, Empresa o Administrador.

Tot seguit es veu un diagrama dels actors del sistema:

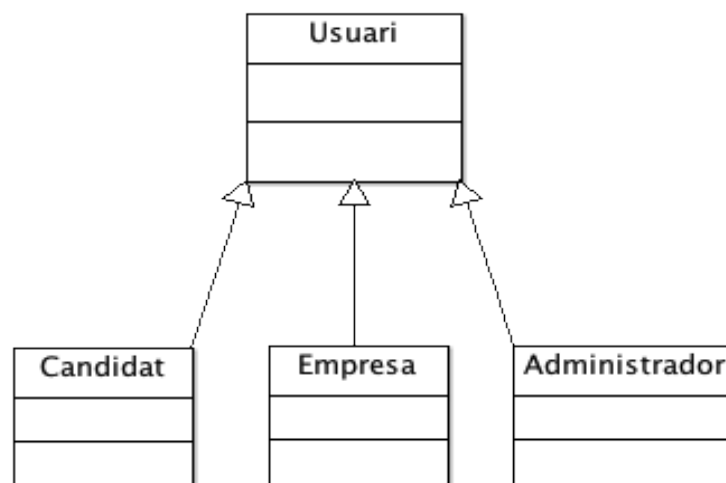


Figura 1: Esquema dels actors del sistema

Restricció textual: Els usuaris de tipus Candidat, Empresa i Administrador han fet "login" a l'aplicació aportant les seves credencials. L'entitat Usuari es correspon amb un usuari no logat.

5. 2 Diagrama de casos d'ús

A continuació es veu el diagrama de casos d'ús.

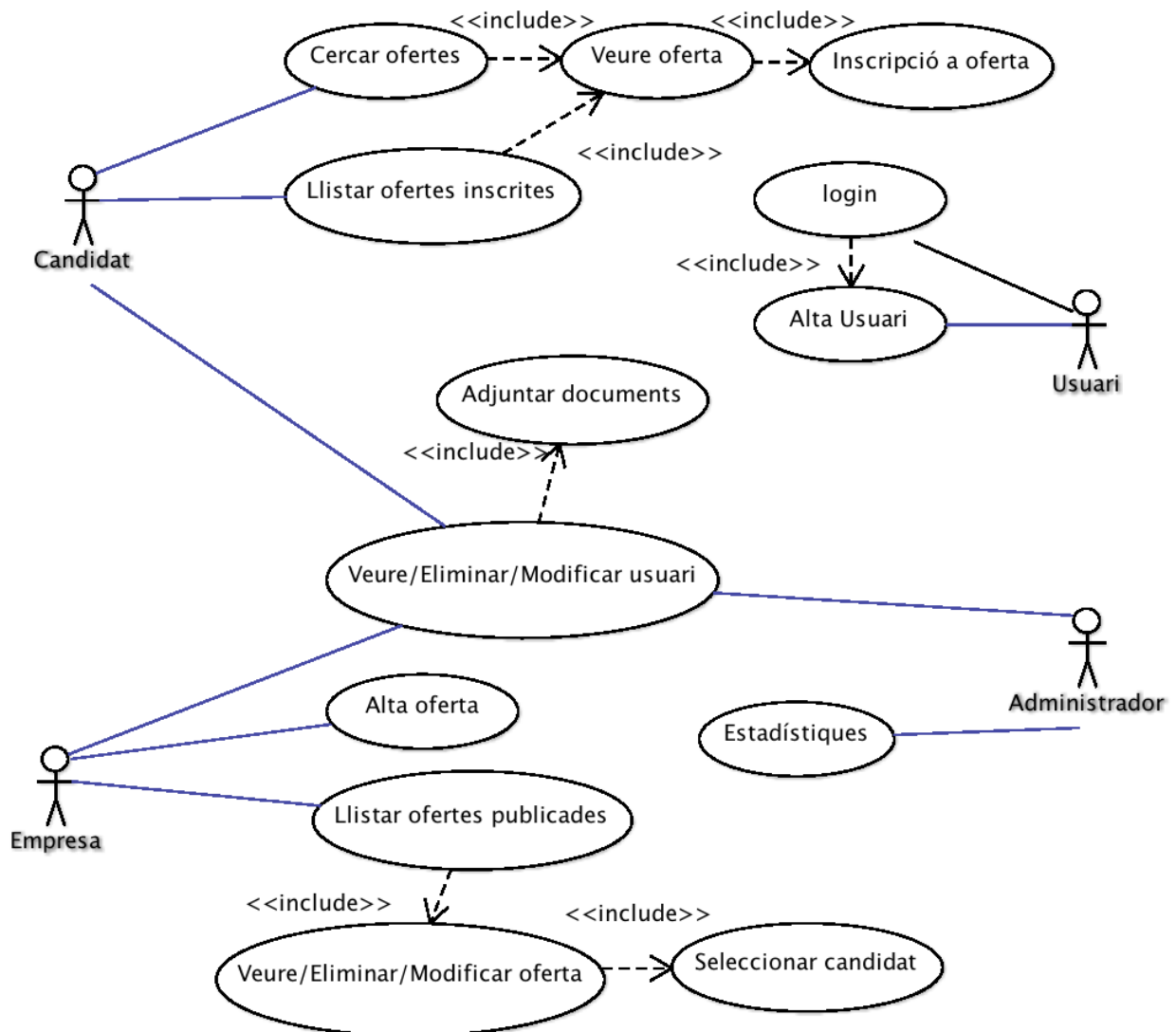


Figura 2: Diagrama de casos d'ús de l'aplicació

5. 3 Especificació de casos d'ús

En aquest apartat es mostra un anàlisi en profunditat dels casos d'ús de l'aplicació. Per esquematitzar-ho, s'han separat els casos d'ús segons la seva tipologia.

5. 3. 1 Casos d'ús: Gestió d'usuaris

5. 3. 1. 1 Alta d'usuari

Cas d'ús: Alta d'usuari al sistema

Actors: Usuari anònim

Descripció: L'usuari accedeix al sistema per tal de crear un compte personal a l'aplicació. Fent això haurà d'especificar diverses dades personals i també el seu tipus: candidat, empresa

Condicció prèvia: Només es podran registrar per aquest cas d'ús els usuaris de tipus candidat i empresa. Els usuaris de tipus Administrador es crearan per un procés diferent.

Curs de l'esdeveniment:

Actor	Sistema
1. L'usuari accedeix a l'aplicació	
	2. El sistema mostra una pàgina genèrica de benvinguda
3. L'usuari clica sobre l'opció "Nou Usuari" per tal de crear el seu compte	
	4. El sistema mostra un formulari en el qual l'usuari ha d'especificar-ne les seves dades
5. L'usuari omple els camps del formulari i prem el botó "Desar"	
	6 El sistema valida les dades i si aquestes són correctes, l'usuari queda desat a la base de dades i redirigeix a l'usuari la pàgina de login

Cas alternatiu:

6.a El sistema detecta que les dades que ha introduït l'usuari són incorrectes i mostra un missatge d'error a l'usuari.

5. 3. 1. 2 Visualització/Modificació de dades d'usuari

Cas d'ús: Visualització i/o modificació de les dades d'usuari

Actors: Usuari de tipus Empresa, Candidat i Administrador

Descripció: L'usuari accedeix a la seva àrea personal per veure, i si s'escau modificar, les seves dades personals.

Condicció prèvia: Cal haver fet login a l'aplicació

Curs de l'esdeveniment:

Actor	Sistema
1. L'usuari accedeix l'apartat d'àrea d'Usuari	
	2. El sistema mostra un formulari que s'omple amb les dades d'usuari actuals.
3. L'usuari pot observar les dades o modificar-les. Un cop editades les dades l'usuari pot clicar el botó Desar.	
	4. El sistema persisteix les dades al sistema i redirigeix l'usuari a la pàgina personal general.

Cas alternatiu:

3.a: L'usuari decideix no modificar cap dada i prem l'opció Cancel·lar.

4.a: El sistema no persisteix les dades i redirigeix l'usuari a la pàgina personal general.

4.b: El sistema detecta que les dades que ha introduït l'usuari són incorrectes i mostra un missatge d'error a l'usuari.

5. 3. 1. 3 Baixa d'un usuari

Cas d'ús: Eliminar les dades d'un usuari

Actors: Usuari de tipus Empresa, Candidat i Administrador

Descripció: L'usuari accedeix a la seva àrea personal per donar-se de baixa de l'aplicació eliminant el seu compte.

Condicció prèvia: Cal haver fet login a l'aplicació

Curs de l'esdeveniment:

Actor	Sistema
1. L'usuari accedeix l'apartat d'àrea d'Usuari	
	2. El sistema mostra un formulari que s'omple amb les dades d'usuari actuals. Aquest formulari inclou un botó d'eliminar usuari.
3. L'usuari prem el botó d'eliminar usuari del sistema.	
	4. El sistema elimina l'usuari del sistema, treu a l'usuari loggat de la sessió i el redirigeix a la pàgina de benvinguda.

Cas alternatiu:

3.a: L'usuari decideix no eliminar el seu usuari i prem l'opció Cancel·lar.

4.a: El sistema no elimina a l'usuari del sistema i el redirigeix a la pàgina personal general.

5. 3. 1. 4 Incorporació de documents

Cas d'ús: Incorporació de documents a Usuaris Candidat.

Actors: Usuari Candidat

Descripció: L'usuari accedeix al sistema per tal d'adjuntar documents amb informació (per exemple CV) al seu compte d'usuari.

Condicció prèvia: Cal haver fet login a l'aplicació i haver accedit a Editar el compte d'usuari (Cas d'ús 5.3.1.2).

Curs de l'esdeveniment:

Actor	Sistema
1. L'usuari observa les seves dades d'usuari obtingudes pel cas d'us 2.3.1.2 i clica a l'opció de Adjuntar documents	
	2. El sistema mostra un formulari on l'usuari pot adjuntar documents
3. L'usuari selecciona els fitxers que desitja adjuntar i prem l'opció d'Adjuntar	
	4. El sistema desa les dades i redirigeix a l'usuari a la seva plana personal.

Cas alternatiu:

3.a L'usuari no vol adjuntar cap document i prem l'opció de cancel·lar.

4.a: El sistema redirigeix l'usuari a la seva pàgina personal.

5. 3. 2 Casos d'ús: Gestió d'ofertes**5. 3. 2. 1 Alta d'ofertes**

Cas d'ús: Publicació d'ofertes de feina

Actors: Usuari de tipus Empresa

Descripció: L'usuari accedeix a la seva àrea personal per tal de publicar una oferta de feina.

Condicció prèvia: Cal haver fet login a l'aplicació

Curs de l'esdeveniment:

Actor	Sistema
1. L'usuari accedeix l'apartat d'àrea personal i prem l'opció de publicar ofertes.	
	2. El sistema mostra un formulari format per camps que l'usuari haurà de complimentar.
3. L'usuari omple les dades requerides i	

prem el botó de publicar.	
	4. El sistema verifica les dades i les emmagatzema al la base de dades.

Cas alternatiu:

3.a: L'usuari decideix no desar les dades i prem l'opció de Cancel·lar

4.a: El sistema no desa les dades i redirigeix a l'usuari a la seva plana personal.

4.b: Les dades introduïdes per l'usuari són incorrectes i mostra un avís d'error a l'usuari.

5. 3. 2. 2 Llistat d'ofertes publicades

Cas d'ús: Llistat d'ofertes publicades

Actors: Usuari de tipus Empresa i Administrador

Descripció: L'usuari accedeix a la seva àrea personal per observar un llistat on apareixen totes les ofertes que ha publicat.

Condicció prèvia: Cal haver fet login a l'aplicació

Curs de l'esdeveniment:

Actor	Sistema
1. L'usuari accedeix l'apartat d'àrea personal i prem l'opció de veure ofertes publicades.	
	2. El sistema mostra un llistat on apareixen les ofertes que ha publicat. Si és buit, no mostra cap element

Cas alternatiu: -

5. 3. 2. 3 Visualització/Modificació d'ofertes

Cas d'ús: Visualització i/o modificació d'ofertes publicades

Actors: Usuari de tipus Empresa i Administrador

Descripció: L'usuari accedeix a la seva àrea personal per veure, i si s'escau modificar, les dades de les ofertes ja publicades.

Condicció prèvia: Cal haver fet login a l'aplicació i haver accedit al llistat d'ofertes (Cas d'ús 5.3.2.2.)

Curs de l'esdeveniment:

Actor	Sistema
1. L'usuari clica sobre el botó Editar corresponent a una oferta ja publicada en el llistat (Veure cas d'ús 2.3.2.2)	
	2. El sistema mostra un formulari amb les dades corresponents a l'oferta seleccionada per l'usuari.
3. L'usuari observa i modifica les dades de l'oferta i prem el botó de Desar	
	4. El sistema persisteix les dades al sistema i redirigeix l'usuari a la pàgina personal general.

Cas alternatiu:

3.a: L'usuari decideix no modificar cap dada i prem l'opció Cancel·lar.

4.a: El sistema no persisteix les dades i redirigeix l'usuari a la pàgina personal general.

4.b: El sistema detecta que les dades que ha introduït l'usuari són incorrectes i mostra un missatge d'error a l'usuari.

5. 3. 2. 4 Baixa d'una oferta

Cas d'ús: Eliminar les dades d'una oferta de feina

Actors: Usuari de tipus Empresa i Administrador

Descripció: L'usuari accedeix a la seva àrea personal per donar-se de baixa una oferta publicada.

Condició prèvia: Cal haver fet login a l'aplicació i haver accedit al llistat d'ofertes publicades (Cas d'ús 5.3.2.2.)

Curs de l'esdeveniment:

Actor	Sistema
1. L'usuari clica sobre el botó Eliminar corresponent a una oferta ja publicada en el llistat (Veure cas d'ús 5.3.2.2)	
	2. El sistema mostra un avís que informa que les dades no seran recuperables.
3. L'usuari clica a Eliminar	
	4. El sistema elimina l'oferta del sistema

Cas alternatiu:

3.a: L'usuari decideix no eliminar l'oferta i prem el botó Cancel·lar.

4.a: El sistema no elimina l'oferta del sistema i el redirigeix a la pàgina personal general.

5. 3. 2. 5 Cerca d'ofertes

Cas d'ús: Cerca d'ofertes de treball

Actors: Usuari de tipus Candidat

Descripció: L'usuari accedeix a la seva àrea personal per observar un llistat on apareixen ofertes de treball segons uns criteris de cerca.

Condició prèvia: Cal haver fet login a l'aplicació

Curs de l'esdeveniment:

Actor	Sistema
1. L'usuari accedeix l'apartat d'àrea personal i accedeix a l'opció de Cerca d'ofertes	
	2. El sistema mostra un formulari on apareixen perquè l'usuari pugui introduir criteris de cerca d'ofertes de feina.

3. L'usuari omple els camps i prem el botó de Cercar	
	4. El sistema mostra les ofertes que coincideixen amb els criteris de cerca

Cas alternatiu:

3.a: L'usuari decideix no cercar i prem el botó Cancel·lar.

4.a: El sistema redirigeix a la pàgina personal general.

5.3.2.6 Inscripció a una oferta

Cas d'ús: Inscripció a una oferta

Actors: Usuari de tipus Candidat

Descripció: L'usuari accedeix a la seva àrea personal per cercar una oferta i inscriure's-hi com a candidat.

Condicció prèvia: Cal haver fet login a l'aplicació i haver cercat les ofertes (Cas d'ús 5.3.2.5)

Curs de l'esdeveniment:

Actor	Sistema
1. L'usuari selecciona una oferta del llistat obtingut a partir dels filtres (Cas d'ús 5.3.2.5)	
	2. El sistema mostra un resum de l'oferta de treball i ofereix l'opció d'inscripció.
3. L'usuari clica sobre l'opció d'Inscripció.	
	4. El sistema desa la inscripció i redirigeix a l'usuari a la seva pàgina personal general.

Cas alternatiu:

3.a: L'usuari decideix no inscriure's a l'oferta i prem Cancel·lar.

4.a: El sistema redirigeix a la pàgina personal general.

5. 3. 2. 7 Llistat d'ofertes inscrites

Cas d'ús: Llistat d'ofertes inscrites

Actors: Usuari de tipus Candidat

Descripció: L'usuari accedeix a la seva àrea personal veure un llistat de totes les seves inscripcions a ofertes.

Condicció prèvia: Cal haver fet login a l'aplicació.

Curs de l'esdeveniment:

Actor	Sistema
1. L'usuari accedeix a la seva pàgina personal i prem l'opció de Veure inscripcions	
	2. El sistema mostra un llistat de totes les ofertes en les que l'usuari s'ha inscrit. Si l'usuari no pertany a cap procés de selecció no es mostrarà cap resultat.

Cas alternatiu: -

5. 3. 2. 8 Selecció de candidat

Cas d'ús: Selecció de candidat

Actors: Usuari de tipus Empresa

Descripció: L'usuari accedeix a la seva àrea personal seleccionar el millor candidat inscrit en cada oferta.

Condicció prèvia: Cal haver fet login a l'aplicació i accedit al llistat d'ofertes publicades (Cas d'ús 5.3.2.2).

Curs de l'esdeveniment:

Actor	Sistema
1. L'usuari prem l'opció de Seleccionar candidat d'una oferta publicada a partir del llistat obtingut (Cas d'ús 5.3.2.2).	
	2. El sistema mostra la llista dels candidats inscrits amb un resum de la seva informació.
3. L'usuari decideix quins dels candidats encaixen millor i els marca com a seleccionats.	
	El sistema desa la informació i redirigeix l'usuari a la seva plana personal general.

Cas alternatiu:

3.a: L'usuari decideix no seleccionar cap candidat i prem l'opció de Cancel·lar..

4.a: El sistema redirigeix a la pàgina personal general.

5. 3. 3 Casos d'ús: Gestió d'accés**5. 3. 3. 1 Login**

Cas d'ús: Login a l'aplicació

Actors: Usuari anònim

Descripció: L'usuari accedeix a la pàgina de login per tal d'accedir a la seva àrea personal de l'aplicació.

Condicció prèvia: Cal haver fet l'alta d'usuari (Cas d'ús 5.3.1.1).

Curs de l'esdeveniment:

Actor	Sistema
1. L'usuari accedeix a la pàgina de login	
	2. El sistema mostra un formulari perquè l'usuari pugui fer login.

3. L'usuari omple els camps amb l'usuari i el password i prem LogIn	
	4. El sistema verifica les dades i si l'usuari existeix redirigeix l'usuari a la pàgina personal.

Cas alternatiu:

3.a: L'usuari introdueix dades incorrectes en el formulari

4.a: El sistema no troba l'usuari en les bases de dades i mostra un missatge d'error.

5. 3. 4 Casos d'ús: Estadístiques**5. 3. 4. 1 Estadístiques**

Cas d'ús: Estadístiques

Actors: Usuari Administrador

Descripció: L'usuari accedeix a la pàgina personal d'estadístiques per tal de veure dades rellevants de l'aplicació.

Condicció prèvia: Cal haver fet login.

Curs de l'esdeveniment:

Actor	Sistema
1. L'usuari accedeix a la pàgina personal d'estadístiques	
	2. El sistema mostra un llistat de possibles estadístiques a executar
3. L'usuari tria una de les estadístiques i prem executar.	
	4. El sistema calcula l'estadística i mostra els resultats a l'usuari.

Cas alternatiu:

3.a: L'usuari no selecciona cap estadística i prem Cancel·lar.

4.a: El sistema redirigeix l'usuari a la seva pàgina personal.

5. 4 Model conceptual

En aquest punt es mostra el diagrama UML corresponent al model conceptual de l'aplicació. En el diagrama es representen les entitats de dades que hauran de ser gestionades en l'aplicació. A continuació es poden observar els diagrames de classes normalitzat i no normalitzat respectivament.

5. 4. 1 Model de dades no normalitzat

En el següent diagrama es pot veure el model de dades previ a la normalització. Com es pot veure, les relacions entre taules no s'han convertit ni en taules ni en claus foranes ni primàries. A part, també es pot veure que l'entitat usuari necessitarà algun tipus per poder diferenciar de quin tipus d'usuari es tracta.

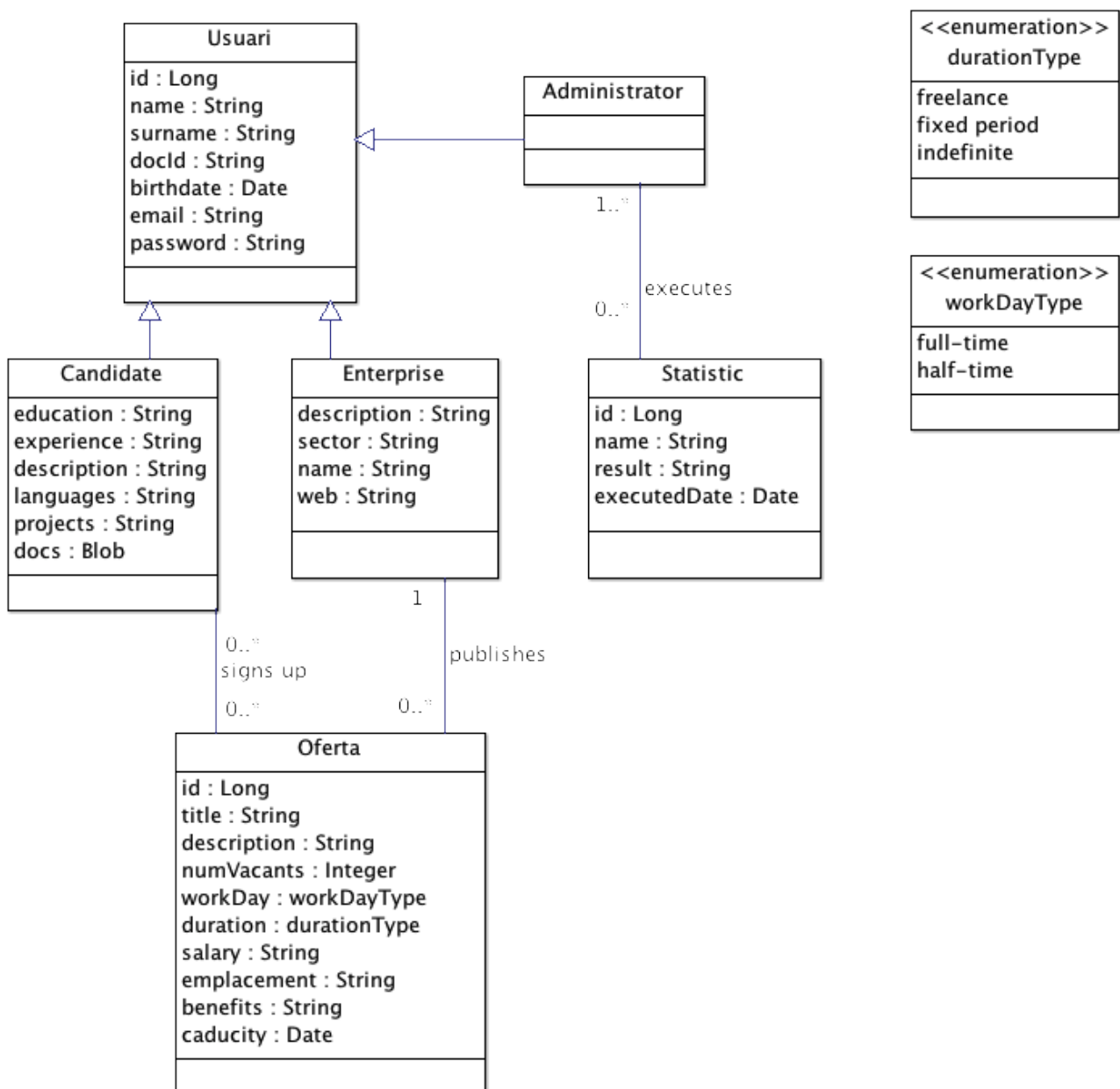


Figura 3: Model conceptual de dades

5. 4. 2 Model de dades normalitzat

En aquest cas, el model de dades ja està normalitzat i està preparat pel següent pas que és la implementació. Si el model de dades s'ha normalitzat correctament, la implementació posterior ha de ser més senzilla i no hi hauria d'haver molts canvis.

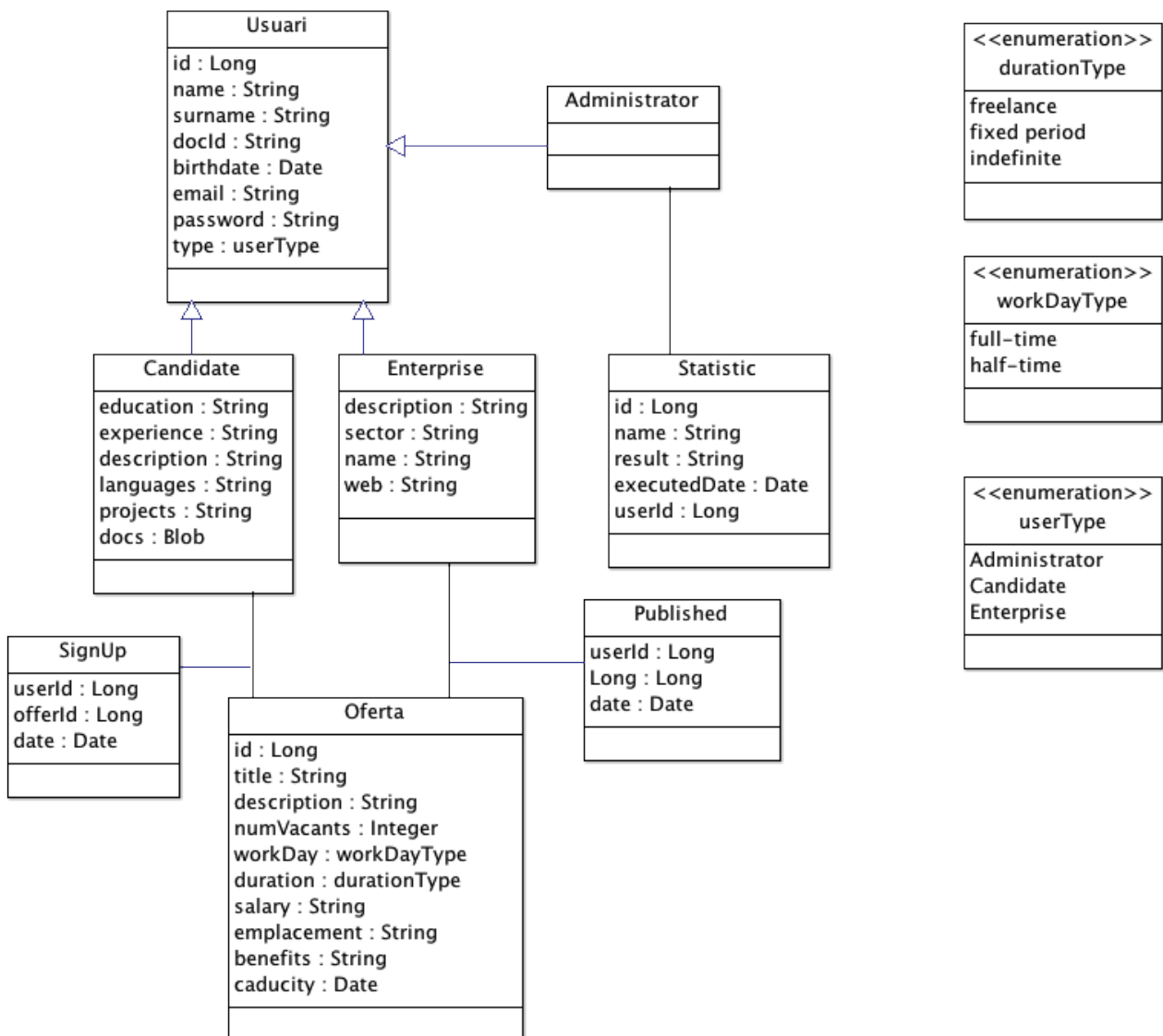


Figura 4: Model conceptual de dades normalitzat

6. Disseny

6. 1 Arquitectura de l'aplicació

L'aplicació web es construeix a partir de play framework (Play!). En aquest detall s'explica més en detall com funciona aquest marc de treball i també és el seu cicle de vida i les característiques més importants.

6. 1. 1 Patró Model-Vista-Controlador

Aquest patró de disseny consisteix en separar l'aplicació en tres capes diferenciades. La idea d'aquest patró és aconseguir, amb aquesta divisió de l'aplicació, un desenvolupament més àgil i també facilitar el futur manteniment.

- **Capa Model:** aquesta capa representa la informació que és específica del domini de l'aplicació. La lògica que apareix en aquesta capa agrega el “significat” a la informació que està plasmada en les dades del sistema. A banda d'això, aquesta capa també inclou una sub-capa específica per a l'accés a les dades.
- **Capa Vista:** o també anomenada capa de Presentació. Aquesta capa desplega la informació del Model de la manera apropiada per a que l'usuari hi pugui interactuar. Play! permet diferents tipus de vistes, com poden ser en format HTML, XML o JSON (per connectivitat amb altres sistemes).
- **Capa Controlador:** és la capa que respon a les accions (normalment generades per l'usuari) i les processa. Aquestes accions es tradueixen en Play! com a simples crides HTTP.

En la següent imatge es pot veure el de manera simplificada l'efecte del patró MVC en el framework Play!. L'usuari genera una petició HTTP (1. HTTP Request) que és gestionada per la capa de Controlador. Aquest s'encarrega de dues tasques. La primera és la de gestionar les dades del Model (2. actualitzant-les, desant noves dades, etc.) i l'altra és la de decidir quina és la vista que caldrà mostrar a l'usuari (3. render). Un cop fet això, el *Controller* retorna a l'usuari tota aquesta informació continguda en la crida HTTP de

resposta (4. HTTP Response).

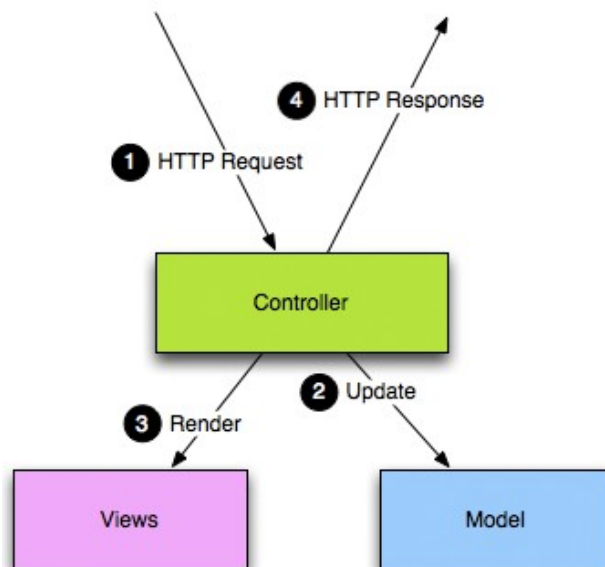


Figura 5: Esquema simplificat de Play Framework

6. 1. 2 Cicle de vida d'una petició HTTP

Com ja s'ha dit anteriorment, Play! és completament stateless (sense estat). Això vol dir que està orientat exclusivament a resoldre peticions/respostes. De fet, es basa en el protocol HTTP que és per definició tampoc te estat.

Més en detall, una petició HTTP segueix el següent procés:

1. L'usuari genera una petició HTTP que és rebuda pel framework
2. El framework detecta aquesta petició i mitjançant un (o més) enrutador detecta quin és el Controlador adequat que ha de tractar la petició i l'invoca. Aquest enrutador disposa del conjunt de rutes http que l'aplicació permet.
3. El Controlador invocat executa l'acció que ha demanat l'usuari i executa la lògica de negoci necessària.
 1. Actualitza les dades del Model (i les recupera si cal mostrar-les a l'usuari).
 2. Identifica la Vista que caldrà mostrar a l'usuari i la renderitza.
4. Amb aquestes dues dades es genera una petició HTTP de resposta cap a l'usuari.

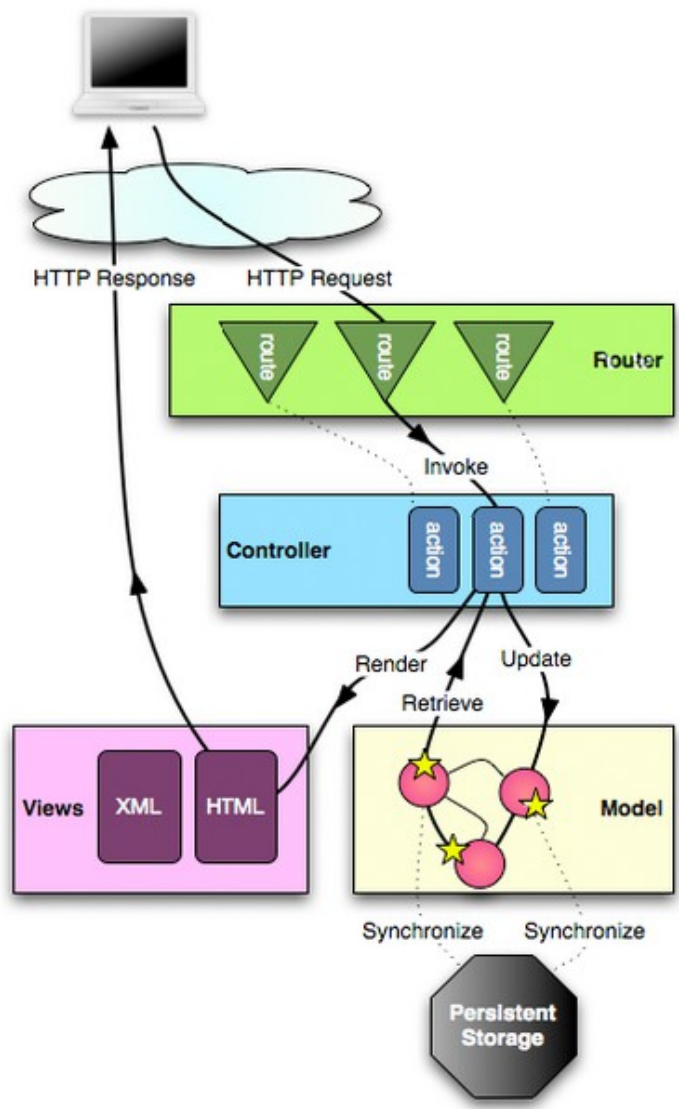


Figura 6: Cicle de vida de play framework

6. 1. 3 Estructura de directoris

Play! organitza els fitxers d'una manera estàndard per tal de mantenir els projectes d'una manera senzilla. A continuació es mostra una taula del que seria l'estructura de directoris bàsica en un projecte amb Play!.

Directoris	Descripció
app	→ Fonts de l'aplicació
└ controllers	→ Controladors
└ models	→ Capa de negoci
└ views	→ Capa de vistes (templates HTML)
build.sbt	→ Script de compilació
conf	→ Configuració i altres
└ application.conf	→ Configuració principal (BD, plugins)
└ routes	→ Fitxer de rutes
public	→ Fitxers públics
└ stylesheets	→ Fitxers CSS
└ javascripts	→ Fitxers Javascript
└ images	→ Fitxers d'Image
project	→ Configuració sbt
└ build.properties	→ Configuració del projecte sbt
└ plugins.sbt	→ Plugins d'sbt
lib	→ Llibreries
logs	→ Directori de logs
└ application.log	→ Fitxer de log per defecte
target	→ Fitxers compilats/generats
test	→ source folder for unit or functional tests

6. 2 Diagrama de seqüència

En aquest punt es mostra un diagrama de seqüència genèric d'una petició HTTP de l'aplicació. En primer lloc es mostra el diagrama d'un cas d'ús simple corresponent a la càrrega de la pàgina principal de l'aplicació. Amb aquest exemple es pretén mostrar el detall de com és una seqüència HTTP en un cas d'ús real.

6. 2. 1 Diagrama d'una petició HTTP

En aquest punt es veu de manera global el procediment de gestió del framework d'una petició HTTP.

- **Fitxer de routes:**

```
# Home page
GET      /                               controllers.Application.index()
```

Com es veu en una part estreta del fitxer de *routes*, apareix una entrada de tipus GET al directori / de l'aplicació. Aquesta petició HTTP demana la càrrega de la plana principal de l'aplicació. Com a correspondència a aquesta petició, es configura el fitxer per tal de el controlador encarregat de gestionar aquesta petició sigui el Controlador *Application*, que està dins el paquet *controllers*. A més a més, també s'indica que cal que s'executi el mètode *index()* d'aquest Controlador, que en aquest cas no rep cap paràmetre.

- **Controlador: controllers.Application**

```
public class Application extends Controller {

    public static Result index() {
        String titleMsg = Messages.get("home.title");
        return ok(index.render(titleMsg, mainmenu.render()));
    }
}
```

Com es pot veure en el controlador anomenat *Application*, apareix un mètode *index()* que es correspon amb l'especificat en el fitxer de rutes. Aquest mètode retorna un objecte *Result* que consisteix en una resposta HTTP. En aquest cas és una resposta satisfactòria (de tipus OK que es correspon amb el codi 200 HTTP) i que rep per paràmetre un objecte *index* que executa un mètode *render*. Això significa que l'objecte *ok* (de tipus *Result*) rep com a paràmetre l'execució del mètode *render* de la vista *index*. (La vista *index* correspon amb el fitxer *index.scala.html* i que conté marcatge HTML5 i sentències Scala). A més a més, el mètode *render* de la vista *index*, a part, rep dos paràmetres més: un text de tipus *String* i també l'execució del mètode *render* d'una altra vista anomenada *mainmenu* (*mainmenu.scala.html*). Amb aquest exemple es pretén il·lustrar com es poden crear vistes formades per altres vistes.

Generalitzant, es pot veure el següent diagrama de seqüència del que seria el cicle de vida d'una petició HTTP amb Play!. (cal tenir en compte que en el cas que es tracta no hi ha aparegut la càrrega de cap objecte del model).

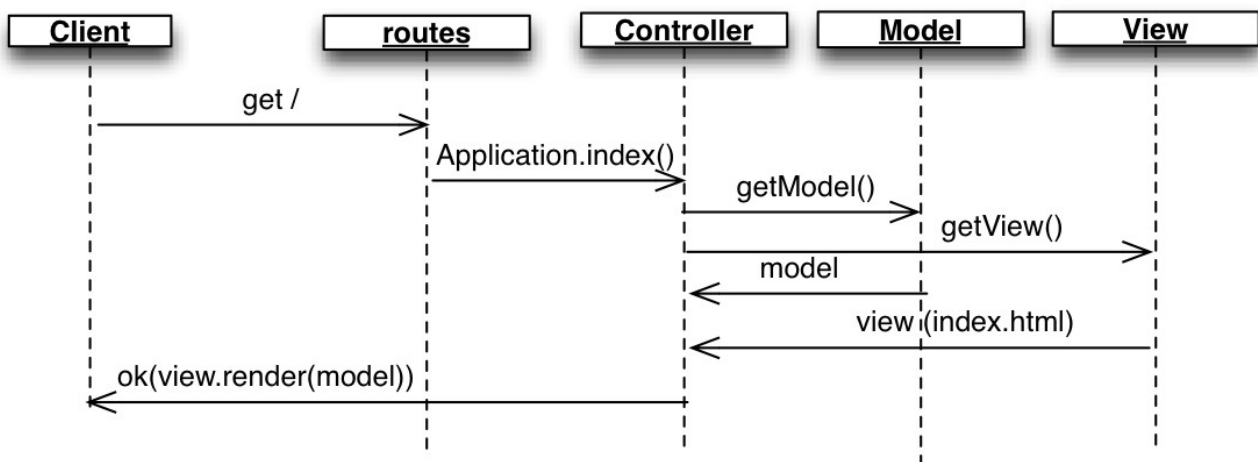


Figura 7: Diagrama de seqüència de petició HTTP Play Framework

- **Model i Views:**

La part del model s'encarregarà de cercar o desar les dades necessàries a la base de dades. Per fer-ho es poden usar classes Java auxiliars a mode de *Helpers* per tal de separar les regles de persistència dels objectes del model. A part, les vistes que caldrà retornar dins la resposta HTTP caldrà que rebin, si s'escau, una sèrie de paràmetres per mostrar dades a l'usuari. Com s'ha pogut veure en el codi del *Controller*, les *views* caldrà

que executin el mètode `render` per tal d'aportar-hi els paràmetres.

Com a curiositat cal dir que les *views* segueixen la convenció d'anomenar-les seguint el patró: *nomVista.scala.html*. Aquest fet és important ja que realment els fitxers contenen marcatge HTML i també sentències Scala. Es podria fer un paral·lelisme amb els fitxers JSP, ja que contindrien també HTML i sentències Java per connectar dades del model amb les vistes. En aquest cas, les vistes Scala funcionen d'una manera anàloga.

```
<div class="span4">
  <h2>@Messages("leftbox.title")</h2>
  <p>@Messages("leftbox.msg") </p>
  <p><a class="btn"
href=@routes.Application.newUser()>@Messages("leftbox.button")</a></p>
</div>
```

Com es pot veure en el codi anterior, en un fitxer `view.scala.html` apareix el marcatge HTML5 i també sentències Scala usant la clau `@`. En aquest cas s'està usant Scala per accedir a dos noms d'etiqueta (*leftbox.title*, *leftbox.msg* i *leftbox.button*) a partir d'un fitxer `.properties` per garantir el multi-idioma, i també executant una crida al mètode `newUser` del controlador `Application` en l'acció de clic d'un botó.

6.3 Configuració de play framework

Playframework disposa de diversos fitxers de configuració que serveixen per parametritzar el comportament de l'aplicació. De totes formes, aquesta configuració és pot efectuar d'una manera simple i sense haver de gestionar fitxers XML molt extensos com pot passar amb altres eines. Tot seguit s'exposa una configuració genèrica d'un projecte Play! i després es donen detalls de paràmetres específics per desenvolupament o producció.

- **Fitxer de routes**

Ja s'ha explicat en part anteriorment. Aquest és un fitxer de configuració que gestiona el fluxe i el tractament de totes les peticions que arriben a l'aplicació per part de l'usuari client. Disposa de tres columnes principals: el tipus de request HTTP (GET, POST, PUT, etc.), la URI o path de la petició i la ruta al controlador i mètode encarregat de gestionar-

la.

```
# Authentication
GET    /login          controllers.Application.login()
POST   /login          controllers.Application.authenticate()
GET    /logout         controllers.Application.logout()
...
```

- **Fitxer application.conf**

Aquest és el fitxer de configuració global de l'aplicació. Permet configurar:

- *Idiomes:*

```
# The application languages
# ~~~~
application.langs="en, ca, en"
```

A part, també caldrà incloure tres fitxers de propietats anomenats *en.properties*, *ca.properties* i *en.properties* que tindran les traduccions en cada idioma respectivament.

- *Bases de dades:*

Es poden configurar diverses connexions a base de dades en paral·lel. Només cal especificar el driver, url, usuari i password de cadascuna de les connexions.

```
#db.default.driver=nom_driver{mysql, Postgresql, oracle, ...}
#db.default.url=URL
#db.default.user=user
#db.default.password=pwd
```

- *Logs:*

Es poden configurar tres nivells de log per l'aplicació (error, info o debug). Cal aportar un fitxer anomenat *application-logger.xml* on es notifica a quin o quins fitxers caldrà escriure les traces del log. A part, aquests es poden activar o desactivar des del fitxer de conf:

```
# Root logger:
logger.root=ERROR

# Logger used by the framework:
logger.play=INFO

# Logger provided to your application:
logger.application=DEBUG
```

6.3.1 Mode desenvolupament

El desenvolupament d'una aplicació amb Play! està pensat per a que sigui senzill i ràpid. Utilitzant la consola de playframework permet al programador des de crear el projecte fins a posar-lo en producció. En aquest punt es veurà la primera part.

Un cop es descarrega Play! només cal descomprimir el fitxer i situar-lo en el path del sistema. Aleshores es podrà usar des de qualsevol directori. Per treballar aleshores:

- Situar-se en un directori per crear un projecte
- Executar la comanda *play new myFirstApp* (amb el nom que es vulgui). Aquest punt permet decidir si volem una aplicació programable amb Scala o Java.
- Un cop creada l'aplicació, podem accedir-hi (*cd myFirstApp*)
- Preparar l'aplicació per treballar amb un IDE. Per exemple, es prepara per poder treballar amb eclipse. Només cal executar la comanda *play eclipse* i després ja es podrà importar fàcilment al IDE.
- Un cop s'hagi desenvolupat, es pot executar l'aplicació desplegant-la en local. Per això es pot usar la comanda *play run*.

Cal notar dues qüestions importants:

- *Servidor embeït Netty²*: És el servidor recomanat tant per desenvolupament i producció. És un servidor creat per Jboss capaç de treballar amb HTTP

2 <http://netty.io/> Netty Server

bidireccional especial per a aplicacions asíncrones i fàcilment escalables.

- *Base de dades en memòria*: Si es vol, es pot optar per usar una base de dades en memòria per no haver de gestionar les dades amb un SGBD³ des del primer moment. Per fer-ho, es pot configurar de la següent manera al fitxer de configuració:

```
db.default.driver=org.h2.Driver
db.default.url="jdbc:h2:mem:play"
db.default.user=sa
db.default.password=""
```

6. 3. 2 Mode producció

Desplegar una aplicació Play! en producció no difereix en excés al mode desenvolupament. El servidor recomanat per Play! a l'hora de desplegar en producció és el mateix Netty embedit a playframework.

En primer lloc cal configurar les bases de dades corresponents al fitxer de configuració. I com a canvi important respecte al mode de desenvolupament, en aquest cas cal executar per consola la comanda *play start* en lloc de *play run*. El mateix servidor realitza uns processos interns diferents per tal de preparar l'aplicació per producció i ja no donar tantes pistes al programador.

6. 3. 2.1 Desplegar a servidors Java EE

Si no es vol fer servir la consola de Play! per executar l'aplicació, existeix l'opció d'empaquetar-la en un war per després desplegar-la a un servidor Java EE tradicional (Jboss, Glassfish, Tomcat, etc.). Per fer-ho només cal executar la comanda *play war nomApp -o nomApp.war*

D'aquesta manera es prepararà un fitxer war capaç d'executar-se en un servidor Java EE.

3 Sistema de Gestió de Bases de Dades (mySQL, Oracle, etc.).

6. 3. 2. 2 Desplegat al núvol (Heroku)

Una altra opció és desplegar l'aplicació en un sistema cloud. La idea és posar l'aplicació en producció deixant de banda els requisits de sistema (els gestionarà el proveïdor de cloud) per centrar-se exclusivament en desenvolupament. Play! és suportat per diversos sistemes cloud, com poden ser Heroku, CloudBees, Cloud Foundry o OpenShift, però en aquest cas es centrarà en Heroku per ser un dels primers sistemes en apostar per Play!.

Heroku és una plataforma cloud PaaS (Platform as a Service) que permet desplegar aplicacions Java (també Node.js, PHP, Scala, Ruby, etc) proveint de tots sistemes necessaris com Bases de dades, gestors de Logs, etc. És un sistema que permet fer-ho de manera gratuïta però a mesura que es necessiten més recursos ja es converteix en una eina de pagament.

Per desplegar l'aplicació cal complir dos requisits: Disposar de Git i Heroku Toolbelt⁴ instal·lats a l'equip.

Passos per desplegar a Heroku:

1. Desar l'aplicació al repositori git local: (per terminal, accedir al directori del projecte)

```
$ git init
$ git add .
$ git commit -m "comentari inicial"
```

2. Crear una aplicació a Heroku:

```
$ heroku create
```

3. Desplegar l'aplicació de Git a Heroku:

```
$ git push heroku master
```

4. El mateix Heroku detecta que l'aplicació que s'està desplegant és de Play! i realitzarà els processos necessaris per tal d'executar-la. Utilitzant la comanda *heroku ps* es pot saber si el deploy ha sigut satisfactori o no.

4 Heroku Toolbelt: Eines per executar comandes Heroku remotes: <https://toolbelt.heroku.com/>

7. Implementació

En aquest apartat es mostra el detall de la implementació del projecte. Per una banda s'enumeren les eines i marcs de treballs usats i per altra es detalla com s'han implementat cadascuna de les capes de l'aplicació.

7.1 Programari utilitzat

Per desenvolupar el projecte s'ha usat el següent, (totes les eines i llibreries tenen llicència Open Source):

- *Java i Scala*: llenguatges de programació sobre JVM (Java Virtual Machine)
- *HTML5 i CSS*: marcatge i disseny de pàgines web
- *Play Framework*: marc de treball REST MVC
- *Twitter Bootstrap*: marc de treball web front-end i responsive
- *Ebean*: implementació de la capa de persistència
- *PostgreSQL i H2*: bases de dades (producció i desenvolupament respectivament)
- *Heroku*: servidor remot de cloud computing
- *Git*: sistema distribuït de repositori de codi
- *GitHub*: repositori públic de codi
- *Eclipse*: entorn de desenvolupament
- *OpenProj*: diagrames de Gantt i planificació
- *LibreOffice*: processador de textos
- *Safari, Firefox i Chrome*: navegadors per desenvolupament i test
- *ArgoUML*: diagrames UML, casos d'ús i seqüència

7. 2 Capa de persistència

Normalment, per implementar la capa de persistència d'una aplicació s'intenta utilitzar algun marc de treball per tal de simplificar la feina de codificació i reutilitzar-ne el codi. Altrament, caldria ocupar-se de gestionar sempre la creació de les connexions a base de dades, gestió de les consultes com també la gestió d'excepcions. Com a punt a favor és que fent-ho d'aquesta manera, sempre se sap què és el que està passant a l'aplicació ja que es poden observar de primera mà les consultes SQL.

Dit això, la idea inicial d'aquest projecte era no treballar directament amb llenguatge SQL tot fent servir marc de treball de tipus ORM (Object Relational Mapping). Inicialment es va pensar en Hibernate, que és un framework que permet relacionar els objectes relacionals Java i els seus atributs amb les taules i columnes respectivament de la base de dades. Això fa que el desenvolupador quedi aïllat dels problemes de les connexions a base de dades, consultes, etc. Com a punt històric, Hibernate va "inspirar" a Java per dissenyar i crear la coneguda JPA (Java Persistence API) com a membre de l'especificació de Java. Hibernate és una bona solució si no es vol entrar en detalls profunds sobre les consultes de l'aplicació. Malauradament, quan un projecte creix ràpidament, és necessari veure amb claredat què és el que està passant amb la base de dades en tot moment, i Hibernate això no ho permet.

Aleshores, al començar a desenvolupar en Play!, es va veure que aquest marc de treball no disposava per defecte ni JPA ni Hibernate, sinó una eina anomenada Ebean. Investigant es va concloure que era una bona solució enfront a JPA i a Hibernate per a usar a l'aplicació. Ebean no deixa de ser un sistema ORM com Hibernate però amb la possibilitat de treballar a nivell SQL en qualsevol moment.

Aquestes raons i el fet d'aprendre a fer funcionar aquesta tecnologia van fer decantar la balança al seu favor. L'aplicació està funcionant sobre Ebean i, a més a més, el sistema permet intercanviar les llibreries per Hibernate o JPA sense haver de modificar excessivament la capa de persistència.

Tot seguit es pot veure el diagrama UML del model relacional extret del sistema de persistència:

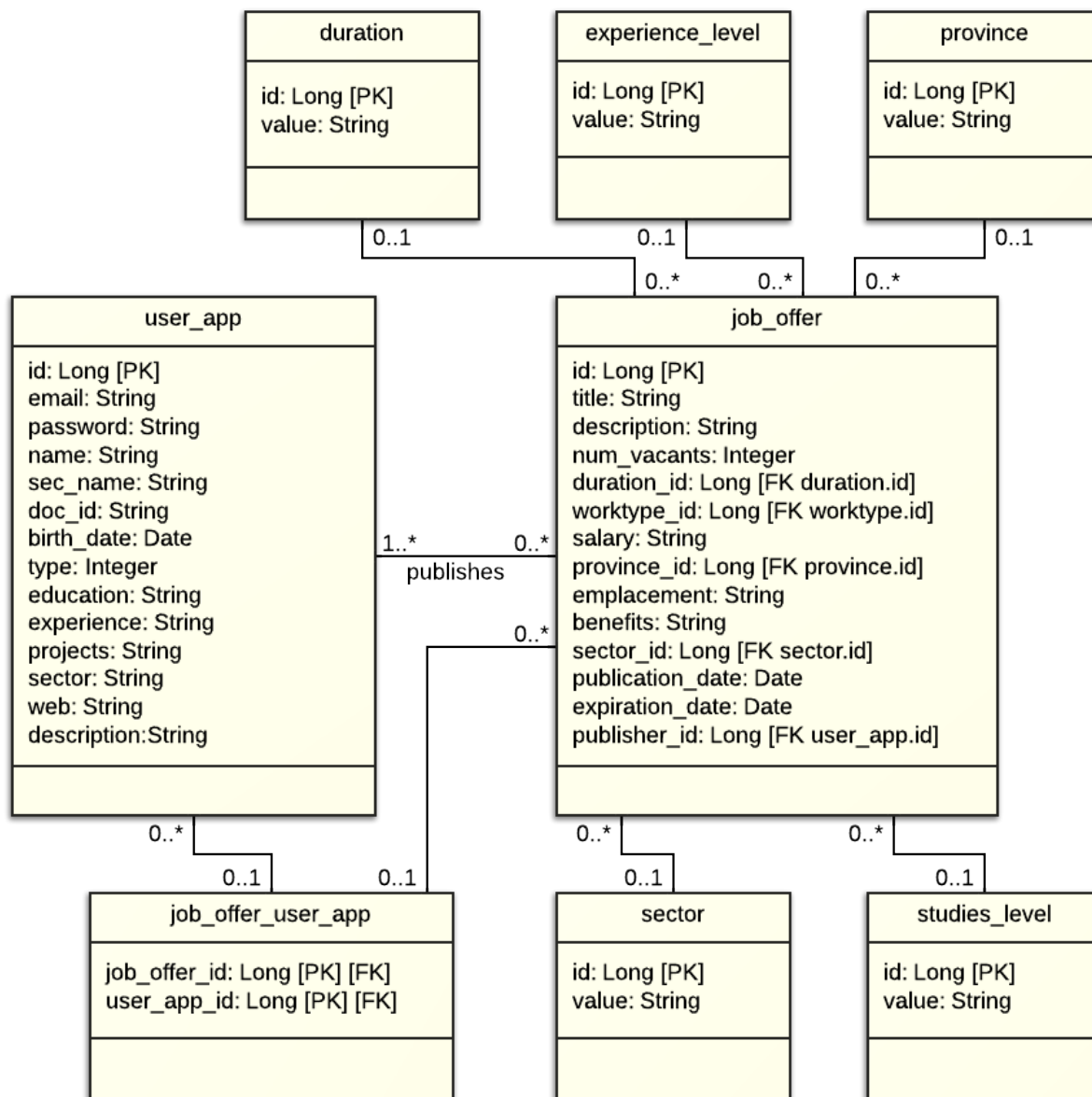


Figura 8: Diagrama UML de la capa de persistència

Per veure les classes Java corresponents al model de dades:

<https://github.com/spawnoi/uocftg/tree/master/app/models>

7.3 Capa de negoci

Un cop vist en l'apartat de disseny el funcionament de les peticions HTTP, tot seguit s'exposa com s'ha organitzat la capa de negoci. Com és sabut, aquesta capa consisteix en el nucli dur de l'aplicació. És on rau la lògica de negoci i és imprescindible que tot segueixi una arquitectura ferma i funcional.

Dit això, cal dir que per implementar aquesta capa s'ha usat un patró de disseny anomenat Façana, que permet agrupar les crides sobre un tipus de funcionalitat en un sol punt. Per exemple, totes les accions que tenen a veure amb *Candidat*, apareixeran en un sol punt (controlador) per a ser gestionades.

Aleshores, de la mateixa manera que el controlador de *Candidate*, s'han creat un seguit de controladors que es corresponen amb *Application* (gestió d'accés), *Secured* (gestió de login), *Company* (gestió de la funcionalitat d'empresa) i *Admin* (gestió de les funcionalitats d'administrador). A banda d'això, també cal tenir en compte que els mètodes d'aquests controladors estan publicats en el fitxer de rutes (routes) per que siguin accessibles via HTTP.

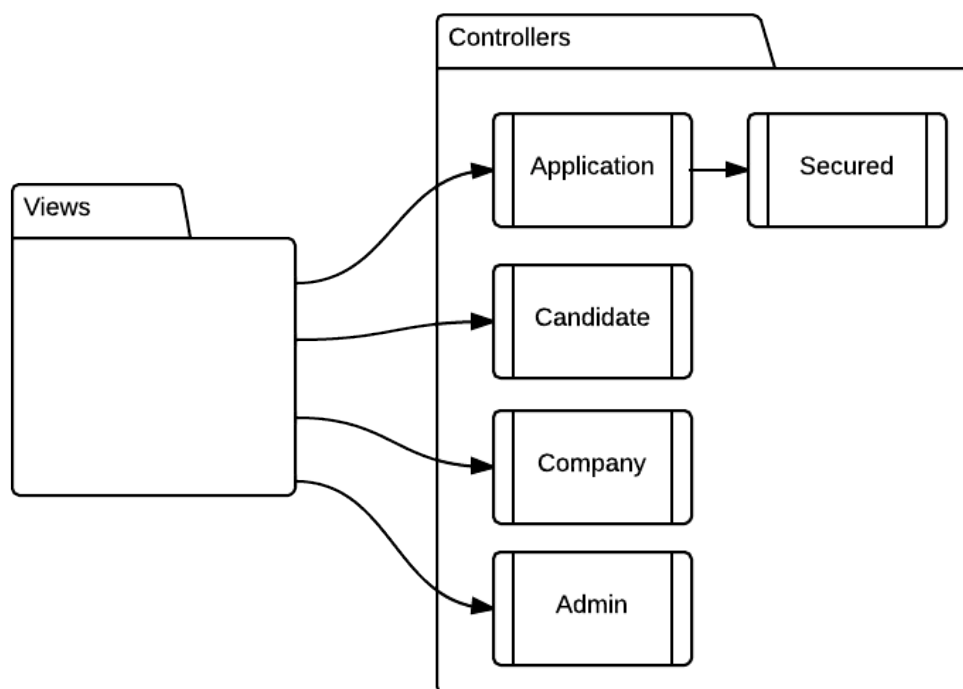


Figura 9: Diagrama del controladors de l'aplicació

Per veure els controladors en detall:

<https://github.com/spawnoi/uoctfg/tree/master/app/controllers>

7. 4 Capa de presentació

La implementació de les vistes ha sigut un treball exigent, ja que ha calgut aprendre en profunditat el marcatge HTML5 i CSS3. Tot i això, per fer-ho s'ha usat una llibreria open source anomenada Twitter Bootstrap (creada per Twitter i alliberada posteriorment).

L'ús d'aquesta llibreria ha fet possible que l'aplicació resultant pugui ser usada des de qualsevol dispositiu connectat a Internet (també anomenat disseny responsive). Com es veu en la imatge del llistat de fitxers de les vistes, s'han separat els fitxers tenint en compte a quina funcionalitat està orientat. Per exemple, apareix la separació entre les vistes corresponents a la part d'administració, de la part d'empresa i finalment la de candidat.

A part, també s'han creat menús específics amb diferents opcions en funció del tipus d'usuari que ha entrat a l'aplicació. Aquesta gestió també ha sigut força complexa però afavoreix molt en la part funcional de l'aplicació.

Per veure en detall les vistes:

<https://github.com/spawnoi/uoctfg/tree/master/app/views>

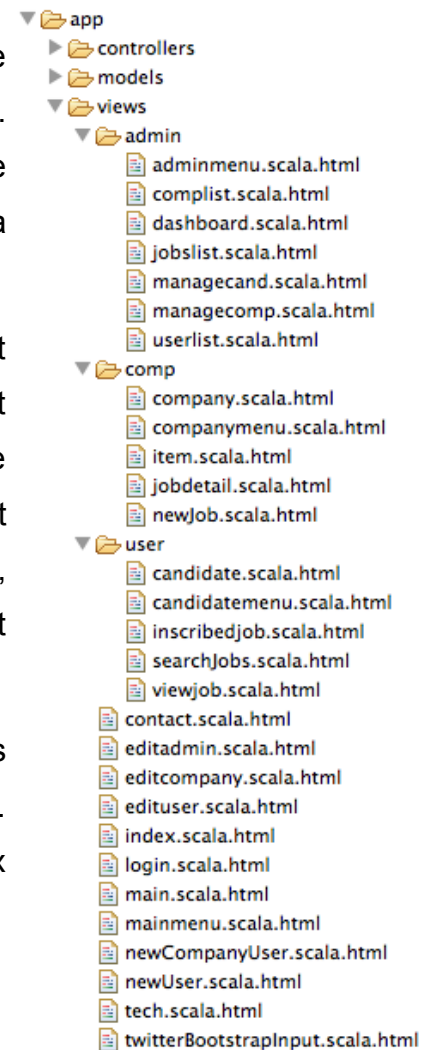


Figura 10: Fitxers de les vistes

7. 5 Verificació

La verificació de l'aplicació s'ha fet mitjançant un seguit d'iteracions. És a dir, per una banda, s'han fet proves a mesura que avançava el desenvolupament. Un cop fet tot el desenvolupament, s'ha fet una segona iteració de proves, tractant tots els requisits, per assegurar la bona implementació d'aquests.

Al ser una aplicació multidispositiu s'ha testejat tant des de navegador (des de la majoria de navegadors moderns com Firefox, Safari o Chrome), i des de dispositius mòbils.

A banda d'això, s'ha proveït d'uns perfils d'usuari de prova per tal de que es pugui entrar a l'aplicació sense la necessitat de crear cap nou usuari. En aquest cas, ja existeixen un seguit de dades de mostra per poder interactuar amb l'aplicació.

En la següent taula es mostren aquests usuaris de prova i les seves claus d'accés:

Usuari (email)	Contrasenya	Perfil
josep.itarte@gmail.com	<i>password</i>	CANDIDAT
josep@sample.com	<i>password</i>	EMPRESA
josep@admin.com	<i>password</i>	ADMINISTRADOR

8. Planificació

Aquest apartat serveix per il·lustrar, en primer lloc, quin va ser el plantejament inicial del projecte, tot planificant unes fases tenint en compte els terminis d'entrega. A part, en aquest punt també s'exposen les tasques o parts de desenvolupament que han impactat en la planificació inicial, tant positivament com al contrari.

8. 1 Planificació inicial

La planificació d'un projecte en el que s'ha de desenvolupar un producte de programari és una tasca complexa i arriscada, ja que normalment aquests projectes acostumen a ser poc predictibles degut a les peculiaritats de les eines software. De totes formes, i tenint en compte les dates d'entrega del projecte (en cadascuna de les fases) es van fer una planificació inicial que sintetitzava els continguts en el següent esquema.

Per estimar la dedicació en hores en cadascuna de les tasques es va considerar que en una setmana s'hi destinaven una hora diària els dies laborables i 4 hores més el cap de setmana. Per tant, això suposa un total de 9h a la setmana que, si es tinguessin en compte només els dies laborals correspondrien un total de 1.8h al dia.

<p><u>Definició formal del projecte (20/09/13 – 02/10/13)</u></p> <ul style="list-style-type: none"> • Descripció del projecte • Objectius i funcionalitats • Descripció de l'arquitectura • Planificació • Documentació Treball final de Grau <p><i>Fita 1. Entrega de la PAC1 – 12/10/13</i></p>	<p>9 jornades</p> <p>16,5 hores</p>
<p><u>Anàlisi i disseny (03/10/13 – 07/11/13)</u></p> <ul style="list-style-type: none"> • Anàlisi de requisits • Especificació <ul style="list-style-type: none"> – <i>Actors del sistema, casos d'ús, model conceptual</i> • Disseny <ul style="list-style-type: none"> – <i>Arquitectura, patrons de disseny, capa de persistència,</i> 	<p>26 jornades</p> <p>47 hores</p>

<p><i>diagrames de seqüència</i></p> <ul style="list-style-type: none"> • Prototipatge de l'aplicació • Documentació Treball final de Grau <p>Fita 2 . Entrega de la PAC2 – 07/11/13</p>	
<p>Implementació (08/11/13 – 16/12/13)</p> <ul style="list-style-type: none"> • Esquema de base de dades • Codificació de l'aplicació <ul style="list-style-type: none"> – Capes Model-Vista-Controlador • Verificació <ul style="list-style-type: none"> – Tests unitaris, cobertura de les necessitats • Documentació Treball final de Grau <p>Fita 3. Entrega de la PAC3 – 16/12/13</p>	<p>27 jornades</p> <p>48,5 hores</p>
<p>Entrega del projecte (17/12/13 – 13/01/14)</p> <ul style="list-style-type: none"> • Revisió de l'aplicació • Revisió de la documentació final • Presentació final <p>Fita 4 . Entrega final del projecte – 13/01/14</p>	<p>20 jornades</p> <p>36 hores</p>

	Nombre	Duració	Inicio	Terminado
1	☐ PAC1 [Descripció formal]	9 days	20/09/13 8:00	2/10/13 17:00
2	Descripció del projecte	1 day	20/09/13 8:00	20/09/13 17:00
3	Objectius i funcionalitat	2 days	23/09/13 8:00	24/09/13 17:00
4	Desc. Arquitectura	3 days	25/09/13 8:00	27/09/13 17:00
5	Planificació	2 days	30/09/13 8:00	1/10/13 17:00
6	Documentació TFG	8 days	20/09/13 8:00	1/10/13 17:00
7	Entrega PAC1	1 day	2/10/13 8:00	2/10/13 17:00
8	☐ PAC2 [Anàlisi i disseny]	26 days	3/10/13 8:00	7/11/13 17:00
9	Anàlisi de requisits	4 days	3/10/13 8:00	8/10/13 17:00
10	Especificació	8 days	9/10/13 8:00	18/10/13 17:00
11	Disseny	7 days	21/10/13 8:00	29/10/13 17:00
12	Prototipatge	5 days	30/10/13 8:00	5/11/13 17:00
13	Documentació TFG	25 days	3/10/13 8:00	6/11/13 17:00
14	Entrega PAC2	1 day	7/11/13 8:00	7/11/13 17:00
15	☐ PAC3 [Implementació]	27 days	8/11/13 8:00	16/12/13 17:00
16	Esquema BD	3 days	8/11/13 8:00	12/11/13 17:00
17	Codificació MVC	19 days	13/11/13 8:00	9/12/13 17:00
18	Verificació	4 days	10/12/13 8:00	13/12/13 17:00
19	Documentació TFG	26 days	8/11/13 8:00	13/12/13 17:00
20	Entrega PAC3	1 day	16/12/13 8:00	16/12/13 17:00
21	☐ Entrega final	20 days	17/12/13 8:00	13/01/14 17:00
22	Revisió aplicació	9 days	17/12/13 8:00	27/12/13 17:00
23	Revisió memòria	5 days	30/12/13 8:00	3/01/14 17:00
24	Presentació final	5 days	6/01/14 8:00	10/01/14 17:00
25	Entrega memòria + codi	1 day	13/01/14 8:00	13/01/14 17:00

Figura 11: Planificació inicial de tasques amb les jornades

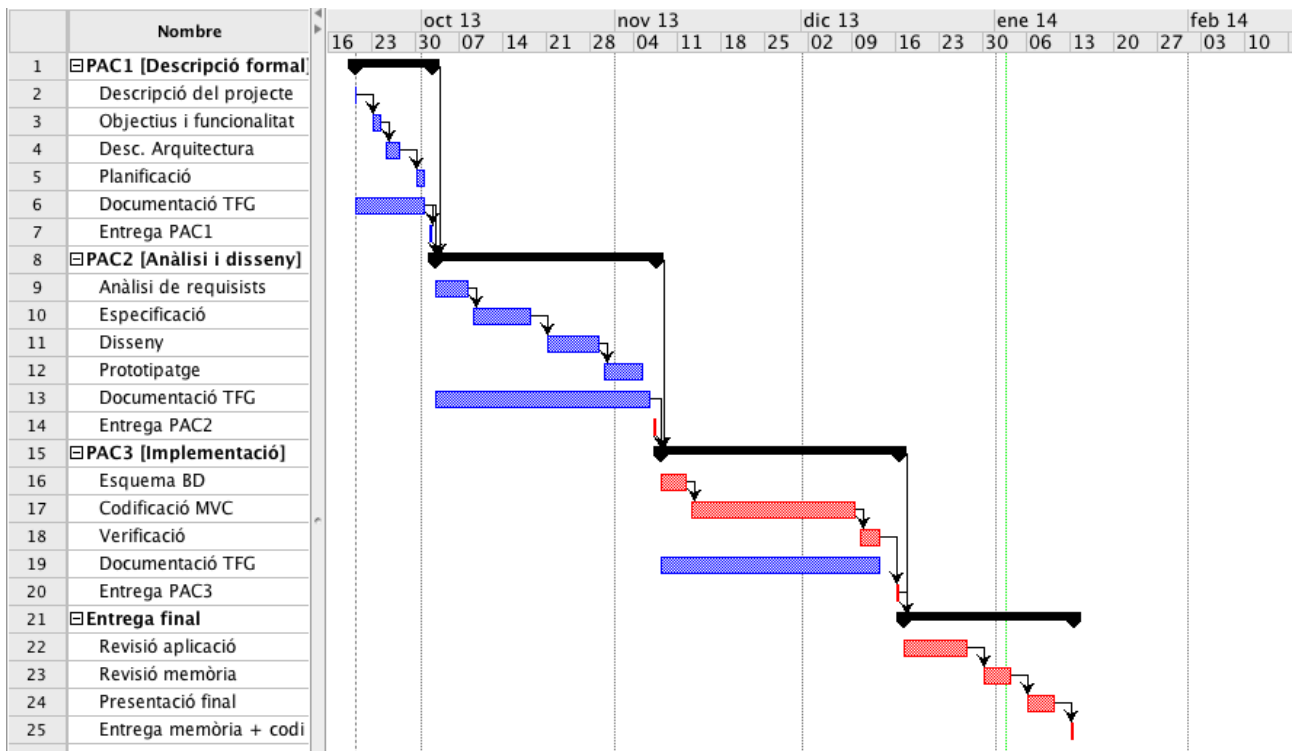


Figura 12: Diagrama de Gantt

8. 2 Planificació final

Un cop acabat el projecte es pot analitzar en quin punt hi ha hagut alguna variació respecte la planificació inicial. Com a factor important que ha afectat en l'estimació inicial es pot esmentar la corba d'aprenentatge pel que fa al disseny i la implementació de la capa de presentació de l'aplicació. Al usar Play! ha calgut aprendre a fer funcionar el motor de vistes usant el llenguatge de programació Scala. A part s'ha integrat el marc de treball Twitter Bootstrap per convertir l'aplicació en multi-dispositiu. Aquest fet també ha incrementat molt les hores destinades a la implementació de les pàgines web.

En resum, els punts que s'han vist més afectats han estat el de Prototipatge i el de Codificació MVC.

En la següent taula es pot veure un resum de les hores estimades respecte a les reals. (Tenint en compte jornades de 1.8 hores com a simplificació de la dedicació setmanal).

Fase	Estimació inicial	Temps real	Impacte
1. Descripció formal	9 dies (16,5 hores)	9 dies (16,5 hores)	0
2. Anàlisi i disseny	26 dies (47 hores)	30 dies (55 hores)	+ 8 hores
3. Implementació	27 dies (48,5 hores)	31 dies (56 hores)	+ 7.5 hores
4. Entrega final	20 dies (36 hores)	20 dies (36 hores)	0

Com es pot veure en la taula hi ha agut un increment de 15,5 hores per motius d'implementació i disseny.

8. 3 Avaluació de costos

Per extreure conclusions més acurades es pot fer un anàlisi de costos en funció de les hores de dedicació. Aquest fet suposa veure el treball com un possible projecte real. Per fer-ho es poden definir uns perfils de treballador, juntament amb els seus preus per hora, i assignar-los les tasques de desenvolupament.

Perfils professionals:

- Gestor de projectes: 60 €/hora
- Analista programador: 35 €/hora
- Tester: 30 €/hora
- Dissenyador: 30 €/hora

Costos en funció dels perfils:

- Gestor de projectes: 10 hores x 60 €/hora → 600 €
- Analista programador: 40 hores x 35 €/hora → 1400 €
- Tester: 15 hores x 30 €/hora → 450 €
- Dissenyador: 30 hores x 25 €/hora → 750 €

Per tant, el projecte ha suposat un cost de 3200 €.

9 Avaluació

Aquest apartat mostra fins a quin punt s'ha arribat en el projecte, quins objectius s'han aconseguit i quins són els propers passos per millorar i ampliar el treball ja fet.

9. 1 Objectius aconseguits

Els objectius, tant els principals com els secundaris, que es detallen en els primers capítols d'aquesta memòria s'han complert.

Com a objectius principals cal fer destacar que s'ha aconseguit crear una borsa de treball plenament funcional que permetrà la comunicació entre empreses i treballadors, fet que intentarà fomentar la contractació donant noves oportunitats. Per altra banda, aquesta borsa de treball està desplegada en un servidor de cloud computing, fet que permet l'accés des de qualsevol dispositiu connectat a Internet.

Els objectius secundaris també s'han satisfet. Com a més destacable trobem que el producte s'ha implementat amb nous marcs de treball i nous llenguatges de programació, encara poc coneguts al nostre país. A part, aquest fet ha permès que hagi calgut aplicar els coneixements apresos durant totes les assignatures cursades al Grau d'Enginyeria Informàtica.

A nivell de requisits funcionals definits en el període d'especificació també han quedat coberts. Resten però alguns requisits no obligatoris per implementar, que com es veurà en el proper punt de treball futur, queden com a properes millores a desenvolupar. Com a requisits no funcionals també queden complerts els fets de que la borsa de treball sigui eficient, segura, usable i disponible en diversos idiomes.

Tot i això, a banda dels objectius i els requisits complerts, també val la pena destacar el caràcter didàctic del treball. Aquest fet ha permet conèixer noves tecnologies i aprofundir en àrees poc conegudes com a desenvolupador fins ara. Entre aquestes cal destacar:

- *Desenvolupament sobre el marc de treball Play framework.* El descobriment d'aquest framework ha sigut molt interessant i convertint el desenvolupament web en Java en una tasca menys complexa i més ràpida.

- *Desenvolupament Web usant HTML5, CSS3 i Scala.* Aquest últim ha sigut necessari per poder usar el motor pàgines que proveeix Play!. A banda, l'ús d'HTML5 i CSS3, de la mà de Twitter Bootstrap, ha permès conèixer de primera mà la complexitat del desenvolupament d'interfícies gràfiques multi-dispositiu.
- *Desplegament de l'aplicació a Heroku.* Aquest fet també ha sigut molt destacable per conèixer quins són els punts clau per poder instal·lar una aplicació web al núvol.
- *Serveis Web REST.* Són inherents al fet d'usar Play!, ja que aquest marc de treball es basa en aquest sistema. Concebre una aplicació web usant aquest sistema també ha sigut molt interessant, sobretot comparant-lo amb altres marcs de treball molt interessants.
- *L'ús de Git i GitHub.* El gestor de versions de fitxers i el repositori públic respectivament han sigut un gran descobriment. Han permès en tot moment disposar del codi actualitzat, segur i disponible en qualsevol moment per descarregar i treballar des de qualsevol dispositiu.

Encara més, tot i havent descobert aquestes eines, també ha calgut refermar els coneixements que ja havia adquirit anteriorment i que ha calgut tornar a aplicar. En primer lloc, el disseny l'aplicació basat en el patró MVC ha sigut clau per mantenir en tot moment una bona arquitectura de desenvolupament. La separació en caps permetrà que en el futur es pugui canviar, per exemple, el gestor de base de dades sense haver de fer canvis significatius al cor de l'aplicació.

9. 2 Treball futur

Com s'ha esmentat en el punt anterior, encara resta treball per fer per disposar d'un sistema encara més funcional i complert. Es va deixar per un futur proper la implementació d'uns requisits poc prioritari. Aquests són els següents:

- *Requisit 5. 3. 1. 4. Incorporació de documents:* es deixa per un futur habilitar la incorporació de documents a la fitxa del candidat.
- *Requisit 5. 3. 4. 1. Estadístiques:* serà interessant ampliar l'oferta d'estadístiques, tant per a usuaris administradors com per a la captació de nous usuaris.

A banda d'aquests requisits, també serà interessant poder implementar unes millores per que el sistema sigui més segur i funcional. Aquestes millores no es van considerar en l'especificació inicial però, un cop treballat en el sistema, serà interessant disposar-les.

- *Encriptació de contrasenyes*: és una mesura de seguretat important i hauria de ser un dels primers canvis a desenvolupar.
- *Comunicació entre candidat i empresa via correu electrònic*: aquest punt també pot ser clau ja que la comunicació serà molt més fluïda i senzilla. A més, es podria gestionar internament permetent al sistema d'estadístiques tractar tota aquesta gestió i explotar-ne les dades.
- *Gestió de la data d'expiració d'ofertes de treball*: això consistirà en deshabilitar l'accés a una oferta de feina si aquesta ja ha caducat.

9.3 Codi font

Com s'ha dit, al llarg del desenvolupament del projecte s'ha fet servir un repositori de codi públic. Aquest sistema ha permès mantenir el codi segur i disponible en qualsevol moment. Concretament s'ha usat GitHub. Si es vol descarregar el codi, hi ha l'opció de “clonar” el projecte (es requereix l'eina Git) o bé descarregar-lo com a fitxer .zip.

En aquest cas, el projecte està disponible a: <https://github.com/spawnoi/uoctfg>

9.4 Aplicació desplegada

L'aplicació resultant al projecte es troba desplegada en un servidor de cloud computing anomenat Heroku. L'adreça per accedir és: <http://arcane-basin-9824.herokuapp.com/>

Aquest sistema, en el seu mode gratuït, deixa les aplicacions en estat de “repòs” quan no hi ha activitat. Per tant, si s'entra per primera vegada i no hi ha activitat al servidor és possible que trigui uns segons més ja que s'està re-arrancant l'aplicació.

10. Conclusions

El desenvolupament d'aquest projecte ha suposat, a part d'un gran esforç i força dedicació, adquirir nous coneixements que segur que seran de gran utilitat en la carrera professional. Aquests coneixements no han estat solament tecnològics, sinó que també s'ha après a com gestionar un projecte de llarga durada intentant no tenir desviacions. Tot seguit apareixen les conclusions extretes un cop finalitzat el projecte.

En primer lloc, el que més ha impactat ha estat el fet de desenvolupar el projecte amb un framework totalment desconegut. Play Framework, com ja s'ha vist al llarg de tota la memòria, ha sigut la pedra angular de la mateixa. Ha calgut força temps per començar-lo, però cal reconèixer que un cop passat aquest temps, desenvolupar és molt més senzill. Treballar amb Play! ha permès veure amb més perspectiva altres frameworks de desenvolupament web en Java com són Spring o Struts, podent veure els punts favorables i desfavorables de cadascun d'ells per poder-los utilitzar en el millor dels casos.

Un altre punt clau del projecte ha estat desenvolupar la capa de presentació usant un framework molt conegut com és Twitter Bootstrap. La idea inicial era crear una aplicació web que fos multi-dispositiu, i amb aquesta eina s'ha aconseguit. Bootstrap soluciona de forma brillant la maquetació de pàgines web obtenint un resultat molt professional. Ha costat també cert temps en dominar aquesta llibreria, sobretot pel que fa als fulls d'estil CSS3. Tot i això, encara hi ha molt camí per recórrer en aquest àmbit per aprofitar-ne al màxim tota la seva potència, per exemple, afegint funcionalitat web usant jQuery.

El projecte també ha sigut útil per aprendre un nou llenguatge de programació. El fet de treballar amb Play! suposa que es pugui usar un motor de vistes (templates) per la capa de presentació. I per usar-les cal fer servir Scala. Descobrir aquest llenguatge de programació ha sigut interessant ja que permetrà que es pugui usar en qualsevol altre projecte. Tot i no vist molt en profunditat, segur que podrà oferir solucions interessants en el futur.

A banda dels aspectes purament tecnològics, també ha sigut destacable el fet de gestionar en primera persona un projecte informàtic. És una tasca complexa planificar un projecte i que, per les dates establertes, entregar un producte acabat que contingui tot el que s'ha proposat inicialment. Aquest fet permet valorar les tasques dels gestors de

projectes, sovint poc agradables i difícils. Completar el projecte observant-ne totes les fases, des de la planificació i especificació inicial, disseny, fins la implementació i verificació ha sigut una experiència molt interessant.

En resum, fer aquest projecte ha suposat que tot l'esforç dedicat s'hagi vist recompensat amb molts nous coneixements adquirits que estic convençut que em serviran en el futur.

11. Glossari de termes

API: Interfície de Programació d'Aplicacions (Application Programming Interface) que conté un conjunt de declaracions que defineix un component informàtic el qual es pot usar. També s'anomena llibreria.

Base de dades: sistema informàtic que manté un conjunt de dades de forma estructurada.

Cloud computing: forma de computació basada en Internet que ofereix els serveis de programari i maquinari com a serveis.

CSS: fulls d'Estil en Cascada (Cascade Style Sheets) és un llenguatge de fulls d'estil per descriure l'aspecte i el format d'algun llenguatge de marcatge (com l'HTML)

Ebean: sistema ORM usat per implementar la capa de persistència d'aplicacions.

Framework: entorn o marc de treball que inclou els recursos o metodologies per tal de desenvolupar ordenadament un projecte de programari.

Heroku: servei de cloud computing basat en Amazon AWS amb disponibilitat de publicació d'aplicacions.

Hibernate: sistema ORM de persistència per gestionar la capa de persistència de les aplicacions.

HTML: llenguatge de marcat molt popular per construir pàgines web.

HTTP: protocol de transferència d'hipertext amb el qual es basa la comunicació a Internet.

Java: llenguatge de programació creat per Sun Microsystems, L'any 2006 es va alliberar com a programari lliure de codi obert.

JPA: Java Persistence API que permet gestionar les connexions a base de dades des de codi Java.

JSON: estàndard obert basat en text dissenyat per l'intercanvi de dades de manera llegible pels humans a través d'Internet.

Login/Logout: el procés pel qual es controla l'accés i la sortida individual a un sistema. Es podria traduir com ingressar/sortir.

LOPD: Llei Orgànica de Protecció de Dades que té com a objectiu garantir i protegir les dades personals en els sistemes informàtics.

Multi-dispositiu: concepte que s'utilitza per determinar que una aplicació està disponible per diverses resolucions, per tant, per diversos dispositius connectats a Internet.

ORM: Mapatge d'Objectes Relacionals (Object-relational mapping) és una tècnica que permet convertir dades del llenguatge orientat a objectes a entitats de bases de dades relacionals.

Play Framework (Play!): framework per construir aplicacions web basades en Java o Scala.

REST, servei web: arquitectura de programari pensada per sistemes distribuïts per tal d'accedir a recursos remots.

Scala: llenguatge de programació funcional que s'executa sobre la Java Virtual Machine (igual que Java).

Twitter Bootstrap: framework per construir la part client d'aplicacions web (i mòbil).

XML: llenguatge de marcatge basat en etiquetes sovint utilitzat per intercanviar dades entre dispositius a través d'Internet.

12. Bibliografia

12. 1 Documentació oficial

Play Framework 2.2.0 API Home. Play Framework [*Índex de la documentació de play!*]
<http://www.playframework.com/documentation/2.2.0/Home>

Scala helpers API. Play Framework [*Documentació de tots els elements Scala que es poden usar en les vistes de play framework*]
<http://www.playframework.com/documentation/2.2.x/api/scala/index.html#views.html.helper.package>

Externalising messages and internationalization. Play Documentation [*Documentació oficial sobre la internacionalització de missatges*]
<http://www.playframework.com/documentation/2.0/Javal18N>

Deploying to Heroku. Play Documentation. [*Informació sobre com desplegar una aplicació Play! a un servidor de cloud computing anomenat Heroku*]
<http://www.playframework.com/documentation/2.2.x/ProductionHeroku>

Heroku Play Framework Support. Heroku Devcenter. [*Documentació oficial de Heroku per desplegar aplicacions Play!*]
<https://devcenter.heroku.com/articles/play-support>

Scala templates common use cases. Play Documentation. [*Documentació oficial sobre l'us de vistes (templates) de play framework*]
<http://www.playframework.com/documentation/2.1.1/ScalaTemplateUseCases>

Bootstrap 2.3.2 API. GetBootstrap. [*Documentació oficial de Bootstrap. Essencial per maquetar pàgines web html5 multidispositiu i conèixer els seus components*]
<http://getbootstrap.com/2.3.2>

Git Documentation. Git. *[Documentació oficial sobre l'eina de control de versions Git, molt interessant per conèixer noves comandes]*

<http://git-scm.com/documentation>

Ebean Documentation. Avajé. *[Documentació sobre el sistema ORM de la capa de persistència. S'hi pot trobar des de documents introductoris com projectes reals amb exemples]*

<http://www.avaje.org/ebean/documentation.html>

12. 2 Articles en línia

How to localise a Play Framework web application. Lunatech blog. *[Informació sobre com convertir una aplicació play! en multi-idioma. Configuració de la banda servidor]*

<http://blog.lunatech.com/2010/04/12/how-localise-play-framework-web-application>

Access translated i18n messages from Scala templates. Stackoverflow *[Informació sobre com obtenir els missatges en diferents idiomes utilitzant les vistes de Play! i Scala]*

<http://stackoverflow.com/questions/10063349/access-translated-i18n-messages-from-scala-templates-play-internationalization>

Hibernate, just stop it! Sickenger. *[Article d'opinió que conté una comparativa entre Hibernate, JPA i Ebean. Interessant per conèixer una mica d'història i poder decidir quina eina és millor en diferents casos.]*

<http://www.sickenger.com/2013/02/hibernate-just-stop-it/>

Play framework and Twtiter's Bootstrap – tutorial. Technology Park. *[Tutorial sobre com integrar en una aplicació Play! el framework de presentació Bootstrap]*

<http://prabhubuzz.wordpress.com/2012/06/27/play-framework-and-twtitters-bootstrap-tutorial-2/>

How to use Twitter Bootstrap 2 with play framework 2.x. Stackoverflow. *[Informació sobre la integració de Bootstrap a Pay!]*

<http://stackoverflow.com/questions/10436815/how-to-use-twitter-bootstrap-2-with-play->

[framework-2-x](#)

Play 2.2 application crashes on Heroku. Stackoverflow. *[Solució a diversos errors a l'hora d'instal·lar aplicacions play a Heroku]*

<http://stackoverflow.com/questions/19391014/play-2-2-application-crashes-on-heroku>

Play! Framework template “include”. Stackoverflow. *[Informació sobre com reutilitzar vistes en diversos fitxers permetent gestionar millor els fitxers esquematitzar les dades.]*

<http://stackoverflow.com/questions/11763912/play-framework-template-include>

Steps needed to use MySQL database with Play framework 2.0. Stackoverflow. *[Informació útil per poder usar MySQL com a gestor de base de dades. També interessant per usar PostgreSQL]*

<http://stackoverflow.com/questions/10007029/steps-needed-to-use-mysql-database-with-play-framework-2-0>

13. Manual d'usuari

En aquest punt es fa un petit recull de les funcionalitats més importants de l'aplicació a mode de manual d'ús. En primer lloc es mostra la pàgina principal:

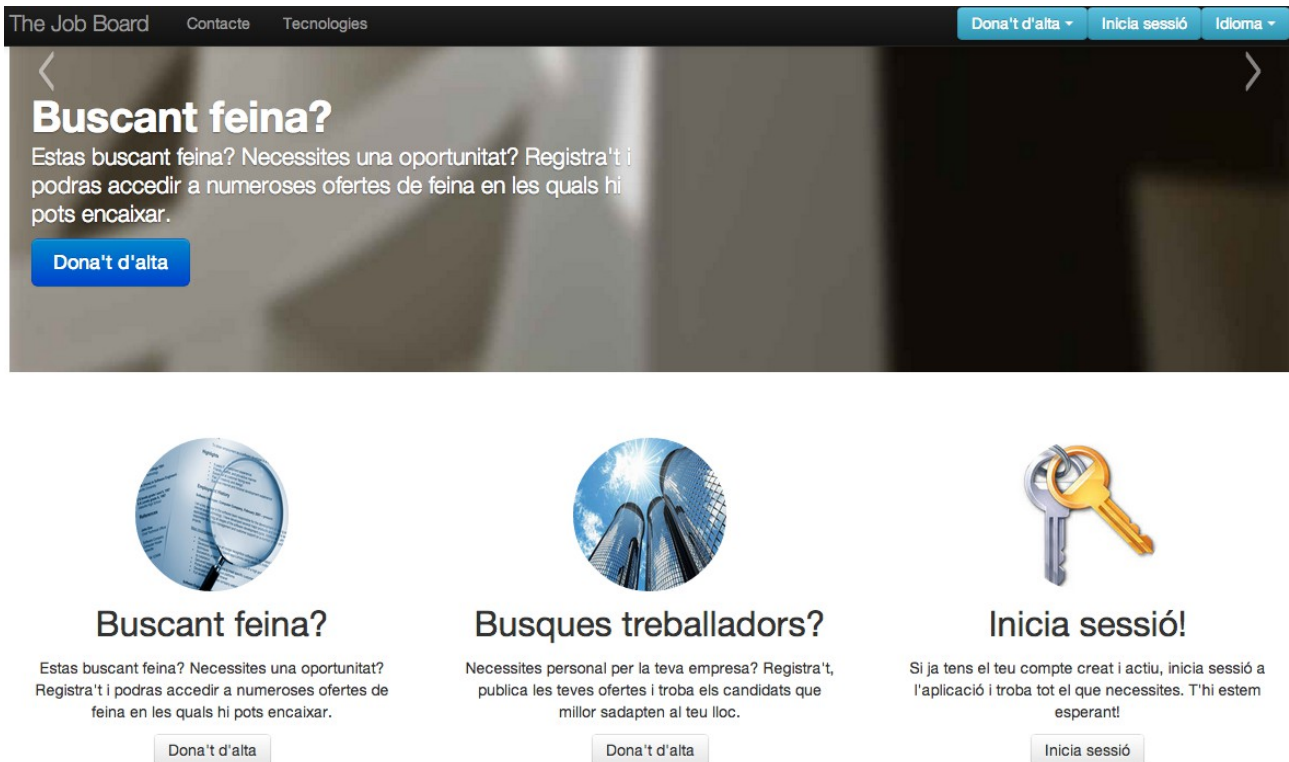


Figura 13: Pàgina principal de l'aplicació (navegador)

En la pàgina principal apareix un menú genèric (sempre que l'usuari no hagi fet login) on consta d'un accés al Home, pàgines de contacte i tecnologies, i les opcions de donar d'alta un nou usuari, iniciar sessió i canviar d'idioma. Actualment l'aplicació està disponible en català, castellà i anglès.



Figura 14: Opcions de nou compte d'usuari



Figura 15: Opcions de canvi d'idioma

Un cop vistes les opcions més importants del menú principal, es poden observar les pàgines de creació d'usuari (de tipus candidat o de tipus empresa).

The image shows a form titled 'Nou perfil candidat'. The form is divided into two columns. The left column contains fields for 'Nom' (Required), 'Cognom' (Required), 'Email' (Email, Required), and 'Contrasenya' (Required). The right column contains fields for 'DNI', 'Educació', 'Experiència', 'Idiomes', and 'Projectes'. At the bottom left of the form, there are two buttons: 'Desa' and 'Cancel·la'. Below the form, there is a footer text: 'Treball Final de Grau d'Enginyeria Informàtica (Eng. Programari) a UOC - Desenvolupat per Josep Itarte - josep.itarte@gmail.com.'

Figura 16: Pàgina de creació d'usuari de tipus Candidat.

The Job Board Contacte Tecnologies [Dona't d'alta](#) [Inicia sessió](#) [Idioma](#)

Nou perfil empresa

Nom	<input type="text"/>	Required	NIF	<input type="text"/>
Lema	<input type="text"/>	Required	Sector	<input type="text"/>
Email	<input type="text"/>	Email, Required	Pàgina web	<input type="text"/>
Contrasenya	<input type="text"/>	Required		

[Desa](#) [Cancel·la](#)

Treball Final de Grau d'Enginyeria Informàtica (Eng. Programari) a UOC - Desenvolupat per Josep Itarte - josep.itarte@gmail.com.

Figura 17: Pàgina de creació d'usuari tipus Empresa

Un cop s'han creat els perfils d'usuari ja es pot accedir a l'aplicació a partir de la pàgina de login.

The Job Board Contacte Tecnologies [Dona't d'alta](#) [Inicia sessió](#) [Idioma](#)

Info! Siusplau, inicia sessió

Inicia sessió

Email	<input type="text"/>
Password	<input type="text"/>

[Inicia sessió](#)

Figura 18: Pàgina d'accés a l'aplicació

Tot seguit es mostren les funcionalitats dels usuaris de tipus Candidat. Si l'usuari ha accedit a l'aplicació apareix directament al llistat d'ofertes on prèviament s'ha inscrit. A part, també apareix un menú concret pel perfil i diferent al genèric. Es poden veure les

opcions d'usuari al menú esquerra o al superior.

The screenshot shows the 'Ofertes inscrites' page. The top navigation bar includes 'The Job Board', 'Ofertes inscrites', 'Cercar ofertes', and a user profile dropdown for 'josep.itarte@gmail.com'. The left sidebar has 'Ofertes inscrites' selected under 'OPCIONES', and 'El meu compte' under 'PERFIL'. The main content area is titled 'Detalls de la oferta' and displays a job listing for 'Java developer using Spring, Hibernate and HTML5' by 'Josep IA softmakers'. A 'Total Inscrits: 2' badge is present. Below the title is a table with job details:

Títol	Java Developer	Núm. vacants	2
Durada	Indefinite	Jornada	Full time
Província	Barcelona	Emplaçament	Barcelona
Salari	30.000€	Beneficis	Access to library
Sector	Informàtica y telecomunicaciones	Publicada	15/12/2013

On the right, there is a 'Desinscriure't' button under the 'Opciones' section.

Figura 19: Pàgina principal de l'àrea Candidat

Quan l'usuari s'ha inscrit en una oferta sempre apareix l'opció de desfer l'acció (Desinscriure't).

The screenshot shows the 'Cercar ofertes' page. The top navigation bar is the same as in Figure 19. The left sidebar has 'Cercar ofertes' selected under 'OPCIONES', and 'El meu compte' under 'PERFIL'. The main content area is titled 'Cercar ofertes' and contains a search form with the following fields:

- Títol:** Text input with 'Java' entered.
- Sector:** Dropdown menu with '-- Escull un sector --'.
- Durada:** Dropdown menu with '-- Escull una durada --'.
- Província:** Dropdown menu with '-- Escull una província --'.
- Emplaçament:** Text input.
- Jornada:** Dropdown menu with '-- Escull una jornada --'.

A 'Cerca' button is located to the right of the search fields. Below the search form is a 'Detalls de la oferta' section, which is identical to the one in Figure 19, showing the same job listing and 'Desinscriure't' button.

Figura 20: Opció de cercar noves ofertes

També hi ha l'opció de que l'usuari pugui canviar les seves dades personals:

The Job Board Ofertes inscrites Cercar ofertes josep.itarte@gmail.com

Àrea de candidat / El meu compte

OPCIONS
Ofertes inscrites
Cercar ofertes

PERFIL
El meu compte

Actualitza el compte personal

Nom Josep	Experiència Years working with Java
Cognom IA	Idiomes Spanish, Catalan and English
DNI 7875690K	Projectes some on github
Educació University	

Actualitza Cancel·la

Figura 21: Actualització del compte d'usuari

Tot seguit es mostren les opcions dels usuaris de tipus Empresa:

The Job Board Ofertes publicades Publicar nova oferta josep@sample.com

Àrea d'empresa / Ofertes publicades

OPCIONS
Ofertes publicades
Publicar nova oferta

PERFIL
El meu compte

Detall de les ofertes

Java developer using Spring, Hibernate and HTML5 Inscrits: 2				Gestiona ▾
— Josep IA softmakers				
Títol	Java Developer	Núm. vacants	2	
Durada	Indefinite	Jornada	Full time	
Província	Barcelona	Emplaçament	Barcelona	
Salari	30.000€	Beneficis	Access to library	
Sector	Informática y telecomunicaciones	Publicada	15/12/2013	

PHP developer with Simphony, HTML5 and CSS experience Inscrits: 1				Gestiona ▾
— Josep IA softmakers				
Títol	PHP Developer	Núm. vacants	3	
Durada	Fixed period	Jornada	Half time	
Província	Barcelona	Emplaçament	Sant Cugat	
Salari	30.000€	Beneficis	no benefits	
Sector	Informática y telecomunicaciones	Publicada	15/12/2013	

Figura 22: Pàgina principal d'usuaris de tipus Empresa

Els usuaris de tipus empresa veuen directament el llistat amb les ofertes que han publicat

amb anterioritat i que encara són actives. A partir d'aquest llistat poden veure els candidats que s'hi han inscrit:

The screenshot shows a job offer detail page. At the top, there is a header 'Detall de les ofertes' and a sub-header 'Opcions'. The main content area displays the job title 'Java developer using Spring, Hibernate and HTML5' with a sub-header '— Josep IA softmakers' and a blue button 'Inscrits: 2'. Below this is a table with the following data:

Títol	Java Developer	Núm. vacants	2
Durada	Indefinite	Jornada	Full time
Província	Barcelona	Emplaçament	Barcelona
Salari	30.000€	Beneficis	Access to library
Sector	Informática y telecomunicaciones	Publicada	15/12/2013

On the right side, under the 'Opcions' header, there is a 'Gestiona' dropdown menu and a 'Veure inscrits' button.

Figura 23: Opció per veure usuaris candidats inscrits

Si s'accedeix a aquesta opció apareix una pàgina on es veu el detall de l'oferta, el total dels inscrits i a la part inferior un llistat on consten les dades dels usuaris candidats inscrits.

The screenshot shows a job offer detail page with a navigation bar at the top: 'The Job Board', 'Ofertes publicades', 'Publicar nova oferta', and a user profile 'josep@sample.com'. Below the navigation bar is a breadcrumb trail: 'Àrea d'empresa / Ofertes publicades / Detalls de la oferta'. On the left side, there are two sections: 'OPCIONS' with links 'Ofertes publicades' and 'Publicar nova oferta', and 'PERFIL' with a link 'El meu compte'. The main content area is divided into two sections: 'Detalls de la oferta' and 'Informació dels inscrits'. The 'Detalls de la oferta' section is identical to the one in Figure 23. The 'Informació dels inscrits' section shows a sub-header 'John' with a sub-header '— Valvich' and a table with the following data:

Nom	John	Email	john@tester.com
Educació	University	Experiència	Experience as a software QA
Idiomes	English, Spanish	Projectes	QA in Amazon.com
Pàgina web	--		

Figura 24: Llistat d'inscrits

A part d'aquestes opcions, l'usuari de tipus empresa també pot publicar les ofertes.

The Job Board Ofertes publicades Publicar nova oferta josep@sample.com

Àrea d'empresa / Publicar nova oferta

OPCIONS
Ofertes publicades
Publicar nova oferta

PERFIL
El meu compte

Publicar nova oferta

Títol <input type="text"/> Required	Província -- Escull una província --
Descripció <input type="text"/> Required	Emplaçament <input type="text"/>
Núm. vacants <input type="text"/>	Durada -- Escull una durada --
Salari <input type="text"/>	Jornada -- Escull una jornada --
Sector -- Escull un sector --	Beneficis <input type="text"/>
	expirationDate <input type="text"/> Date ('dd/MM/yyyy')

Desa Cancel·la

Figura 25: Pàgina per publicar noves ofertes

I com s'ha vist amb l'usuari de tipus candidat, també pot modificar les seves dades d'empresa.

The Job Board Ofertes publicades Publicar nova oferta josep@sample.com

Àrea d'empresa / El meu compte

OPCIONS
Ofertes publicades
Publicar nova oferta

PERFIL
El meu compte

Actualitza el compte personal

Nom <input type="text"/> Josep IA softmakers	NIF <input type="text"/> 123123123k
Lema <input type="text"/> We make software for you!	Sector <input type="text"/> Soft. Development
	Pàgina web <input type="text"/> http://fk-enterprise.com

Actualitza Cancel·la

Figura 26: Actualització de perfil d'usuari empresa

Tot seguit es mostra la pàgina principal de l'usuari de tipus Administrador:

The Job Board Dashboard Gestiona candidats Gestiona empreses Gestiona ofertes josep@admin.com

Àrea d'administrador / Dashboard

OPCIONS
Dashboard
Gestiona candidats
Gestiona empreses
Gestiona ofertes

PERFIL
El meu compte

Candidats (6)
Accedeix a la gestió de candidats on podràs actualitzar les dades o eliminar els candidats incorrectes
[Gestiona candidats](#)

Empreses (2)
Accedeix a la gestió d'empreses on podràs observar totes les empreses i modificar o adaptar-ne les dades
[Gestiona empreses](#)

Ofertes (4)
Accedeix a la gestió d'ofertes de treball, observa-les a partir d'un llistat i elimina les ofertes expirades
[Gestiona ofertes](#)

El meu compte
Desconnecta't

Figura 27: Pàgina principal de l'usuari de tipus Administrador

Des d'aquesta pàgina principal es pot accedir a les opcions d'administrador.

The Job Board Dashboard Gestiona candidats Gestiona empreses Gestiona ofertes josep@admin.com

Àrea d'administrador / Gestiona candidats

OPCIONS
Dashboard
Gestiona candidats
Gestiona empreses
Gestiona ofertes

PERFIL
El meu compte

Gestiona candidats

Filtra Filtra

Nom	Cognom	Email	DNI	Data naixement	Opcions
Georgina	Patterson	user@patterson.com	342343	-	Gestiona ▾
James	White	jwhite@candidate.com	42342	-	Elimina Actualitza
Joan	Capde	jc@capde.com	23324231K	-	Gestiona ▾
John	Valvich	john@tester.com	9876543L	-	Gestiona ▾
Josep	IA	josep.itarte@gmail.com	7875690K	-	Gestiona ▾
Oriol	Malet	cand@malet.com	949384	-	Gestiona ▾

← Previ Mostrant 1. a .6. de .6 Següent →

Figura 28: Administrador d'usuaris candidats del sistema

The Job Board Dashboard Gestiona candidats Gestiona empreses Gestiona ofertes Josep@admin.com

Àrea d'administrador / Gestiona empreses

OPCIONS
 Dashboard
 Gestiona candidats
Gestiona empreses
 Gestiona ofertes

PERFIL
 El meu compte

Gestiona empreses

Filtra Filtra

Nom	NIF	Email	Sector	Pàgina web	Opcions
Better-Soft	7876	info@bettersoft.com	Soft Dev and WA	bettersoft.com	Actualitza
Josep IA softmakers	123123123k	josep@sample.com	Soft. Development	http://fk-enterprise.com	Actualitza

← Previ Mostrant 1. a .2. de .2 Següent →

Figura 29: Administració d'empreses del sistema

The Job Board Dashboard Gestiona candidats Gestiona empreses Gestiona ofertes Josep@admin.com

Àrea d'administrador / Gestiona ofertes

OPCIONS
 Dashboard
 Gestiona candidats
 Gestiona empreses
Gestiona ofertes

PERFIL
 El meu compte

Gestiona ofertes

Filtra Filtra

Títol	Núm. vacants	Inscrits	Publicada	Expira	Opcions
Java Developer	2	2	15 Dec 2013	03 Feb 2014	Elimina
PHP Developer	3	1	15 Dec 2013	-	Elimina
QA Senior	1	2	15 Dec 2013	01 Feb 2014	Elimina
Videogames QA	2	1	15 Dec 2013	02 Feb 2014	Elimina

← Previ Mostrant 1. a .4. de .4 Següent →

Figura 30: Gestió de les ofertes de treball del sistema

I a part de les opcions pròpies del perfil Administrador, com la resta de perfils també pot editar les seves dades personals.

The Job Board Dashboard Gestiona candidats Gestiona empreses Gestiona ofertes josep@admin.com

Àrea d'administrador / El meu compte

OPCIONS
 Dashboard
 Gestiona candidats
 Gestiona empreses
 Gestiona ofertes

PERFIL
 El meu compte

Actualitza el compte personal

Nom: Josep Admin

Pàgina web: http://admin-web.com

Cognom: Adm

Actualitza Cancel·la

Figura 31: Actualització de dades d'Administrador

Tot i les funcionalitats bàsiques de l'aplicació, també apareixen un parell de pàgines a nivell informatiu. Una d'elles conté informació sobre les tecnologies emprades en el desenvolupament i l'altra conté dades de contacte:

The Job Board Contacte Tecnologies josep@admin.com Idioma

Aquesta aplicació s'ha desenvolupat amb les següents tecnologies!

Nom	Tipus	Informació
Java	Llenguatge de Programació	http://java.com
Scala	Llenguatge de Programació	http://www.scala-lang.org/
HTML5	Marcatge HTML	http://www.html5rocks.com/en/
CSS3	Disseny web	http://www.w3schools.com/css/
Twitter Bootstrap 2.3.2	Front-end responsive web framework	http://getbootstrap.com/2.3.2/
Play Framework	MVC REST Web framework	http://www.playframework.com/
EBeans	Capa Persistència	http://www.avaje.org/
PostgreSQL	Motor de base de dades	http://www.postgresql.org/
Heroku	Cloud computing	https://www.heroku.com/
Git	Sist. distribuït repositori de codi	http://git-scm.com/
GitHub	Gestió de repositoris públics	http://github.com

Figura 32: Pàgina sobre les tecnologies utilitzades

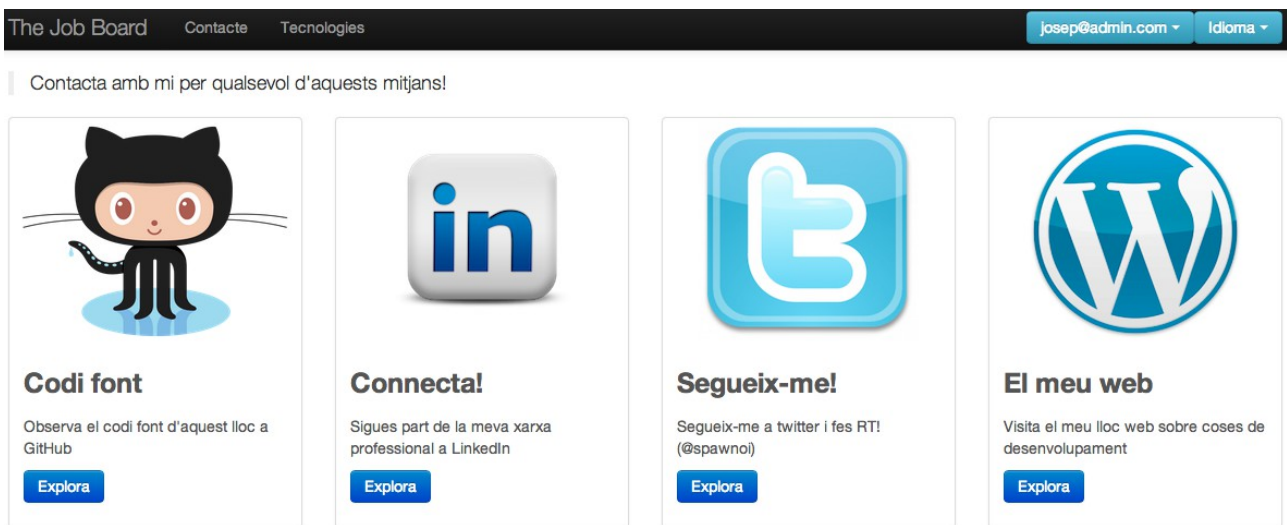


Figura 33: Pàgina amb informació de contacte

Al ser una aplicació multi-dispositiu, tot seguit es mostra una imatge de l'aplicació adaptada a una mida de dispositiu inferior, com podria ser una tauleta.



Àrea d'administrador / Gestiona ofertes

OPCIONES

- Dashboard
- Gestiona candidats
- Gestiona empreses
- Gestiona ofertes**

PERFIL

- El meu compte

Gestiona ofertes

Filtra Filtra

Títol	Núm. vacants	Inscrits	Publicada	Expira	Opcions
Java Developer	2	2	15 Dec 2013	03 Feb 2014	Elimina
PHP Developer	3	1	15 Dec 2013	-	Elimina
QA Senior	1	2	15 Dec 2013	01 Feb 2014	Elimina
Videogames QA	2	1	15 Dec 2013	02 Feb 2014	Elimina

Figura 34: Aplicació adaptada a tauleta