

# **Solució J2EE per a la gestió de la mobilitat**

*MobiDroid: Gestor de Mobilitat*

**Jordi Pérez Gómez**

ETIG

**Joan Vicent Orenge Serisuelo**

18 Juny 2012



A Leo, per la teva estima, paciència i sacrifici durant tots aquests mesos.

A la meva família, però en especial a la meva mare,  
que sé que estaràs molt orgullosa i t'ho mereixes tot.

## RESUM DEL TREBALL FI DE CARRERA

El projecte **MobiDroid** neix de la inquietud per aprofundir gran part dels coneixements adquirits durant els estudis d'ETIG amb els sistemes distribuïts existents a l'actualitat, seguint els estàndards del **J2EE**, nom de l'àrea TFC escollida.

Una vegada decidit el projecte, i ja en plena fase de planificació, la primera sensació és de vertigen, quan prenc consciència que saber programar en Java no és ni molt menys suficient per a implementar tot allò que pretenc incloure: Struts2, MVC, Hibernate, Tomcat, html, JSP, tiles, Ajax...

A tot aquest conjunt de conceptes i llenguatges, li afegeixo com a punt molt important un client Android que donarà molt de sentit a la finalitat del producte a desenvolupar:

- Una **agenda col-laborativa** on els treballadors disposin i actualitzin la informació necessària per a dur el seguiment de les seves tasques.

Juntament amb el client Android, he trobat convenient tenir un client web que s'encarregarà de la part de gestió i manteniment del producte.

Així doncs, tinc una oportunitat molt bona de posar en pràctica molts dels conceptes que tant es remarquen en els estudis d'ETIG, i seguint aquest pretext pretenc construir una aplicació modular, amb components reutilitzables i amb un alt grau d'escalabilitat per a fer evolucionar les funcionalitats o incorporar-ne de noves.

No cal dir que ha estat un repte enorme endinsar-me en un projecte d'aquesta complexitat i en tan poc temps, en el que ha calgut estudiar, digerir i posar en pràctica molts conceptes nous per mi. En aquesta memòria tractaré de posar de manifest tota aquesta experiència.

## INDEX

<b>1.</b>	<b>INTRODUCCIÓ</b>	<b>6</b>
1.1.	JUSTIFICACIÓ DEL TFC	6
1.2.	OBJECTIUS DEL TFC	7
1.2.1.	CLASSIFICACIÓ DELS OBJECTIUS	7
1.3.	ENFOCAMENT I MÈTODE SEGUIT	8
1.4.	PLANIFICACIÓ DEL PROJECTE	9
1.5.	PRODUCTES OBTINGUTS	12
1.6.	CONTINGUT DELS ALTRES PUNTS DE LA MEMÒRIA	13
<b>2.</b>	<b>PRESA DE REQUERIMENTS</b>	<b>14</b>
2.1.	RESTRICCIONS	14
2.2.	NORMATIVA I LEGISLACIÓ	14
2.3.	AVALUACIÓ DE RISCOS	14
2.4.	PLA DE CONTINGÈNCIA	15
<b>3.</b>	<b>ESPECIFICACIONS FUNCIONALS</b>	<b>16</b>
3.1.	VISIÓ GENERAL	16
3.2.	GESTIÓ DE MOBI DROID. EL CLIENT WEB	17
3.2.1.	OBJECTIUS I REQUISITS	17
3.2.2.	PERFILS D'AUTORITZACIÓ	18
3.2.3.	FUNCIONALITATS DEL CLIENT WEB	19
3.2.3.1.	MENÚ PERSONALITZAT PER A CADA PERFIL D'USUARI	19
3.2.3.2.	AGENDA PERSONAL	20
3.2.3.3.	GESTIÓ DE L'AGENDA	20
3.2.3.4.	PERFIL DE L'USUARI	21
3.2.3.5.	MANTENIMENT D'USUARIS	21
3.2.3.6.	MANTENIMENT DE CLIENTS I CONTACTES	22
3.2.4.	CASOS D'ÚS	23
3.2.4.1.	CONNEXIÓ	24
3.2.4.2.	MANTENIMENT	26
3.2.4.3.	AGENDA	29
3.2.5.	PROTOTIPUS DE PANTALLES	30
3.2.5.1.	BENVINGUDA I AUTENTICACIÓ	30
3.2.5.2.	AGENDA	30
3.2.5.3.	DETALL DE L'AGENDA (CREAR, EDITAR)	31
3.2.5.4.	PERFIL PERSONAL	31
3.2.5.5.	MANTENIMENT D'USUARI	32
3.2.5.6.	DETALL DE L'USUARI	32
3.2.5.7.	MANTENIMENT DE CLIENT	32
3.2.5.8.	DETALL DE CLIENT	32
3.2.5.9.	DETALL DE CONTACTE	33
3.2.6.	PROPOSTES DE MILLORA	33
3.3.	MOBI DROID PER A DISPOSITIUS MÒBILS. EL CLIENT ANDROID	33
3.3.1.	OBJECTIUS I REQUISITS	33
3.3.2.	PERFILS D'AUTORITZACIÓ	34
3.3.3.	FUNCIONALITATS DEL CLIENT ANDROID	34
3.3.3.1.	NAVEGACIÓ PER LES PANTALLES	34
3.3.3.2.	AGENDA PERSONAL	35
3.3.3.3.	MODIFICACIONS A L'AGENDA PERSONAL	35
3.3.3.4.	EXPLOTACIÓ DE LA INFORMACIÓ	36
3.3.4.	CASOS D'ÚS	36
3.3.4.1.	CONNEXIÓ	37
3.3.4.2.	AGENDA	38
3.3.5.	PROTOTIPUS DE PANTALLES	40
3.3.5.1.	BENVINGUDA I AUTENTICACIÓ	40
3.3.5.2.	AGENDA (FORMAT TAULETA)	40
3.3.5.3.	AGENDA (FORMAT MÒBIL/VERTICAL)	41
3.3.6.	PROPOSTES DE MILLORA	42

<b>4.</b>	<b>ESPECIFICACIONS TÈCNIQUES .....</b>	<b>43</b>
4.1.	ARQUITECTURA DEL PROJECTE .....	43
4.1.1.	J2EE .....	43
4.1.2.	APACHE TOMCAT .....	44
4.1.3.	MYSQL SERVER.....	44
4.1.4.	HIBERNATE .....	44
4.1.5.	APACHE STRUTS2.....	44
4.1.6.	APACHE TILES .....	45
4.1.7.	SDK ANDROID.....	45
4.1.8.	ECLIPSE .....	45
4.1.9.	VERSIONS .....	45
4.2.	DISSENY DE LA PERSISTÈNCIA.....	46
4.2.1.	ATRIBUTS .....	47
4.3.	PROJECTE WEB .....	48
4.3.1.	RECURSOS .....	48
4.3.2.	ESTRUCTURA DE PAQUETS .....	48
4.3.3.	DIAGRAMES DE CLASSES .....	49
4.3.3.1.	DIAGRAMA DE CLASSES D'ENTITAT .....	49
4.3.3.2.	DIAGRAMA DE CLASSES DAO.....	50
4.3.4.	IMPLEMENTACIÓ D'STRUTS2.....	51
4.3.4.1.	PATRÓ MVC .....	52
4.3.4.2.	JSP I TILES.....	53
4.3.4.3.	ESTILS CSS.....	53
4.3.4.4.	INTERNACIONALITZACIÓ (i18N) .....	53
4.3.4.5.	CONTROL DE SESSIÓ DEL CLIENT WEB .....	54
4.3.4.6.	VALIDACIONS A CLIENT I A SERVIDOR .....	54
4.4.	PROJECTE ANDROID .....	55
4.4.1.	RECURSOS .....	55
4.4.2.	ESTRUCTURA DE PAQUETS .....	55
4.4.3.	SERVEIS WEB. JSON.....	56
4.4.4.	COMPOSICIÓ DELS LAYOUTS .....	57
4.4.5.	INTERNACIONALITZACIÓ (i18N).....	57
4.5.	VALORACIÓ ECONÒMICA.....	58
4.6.	CONCLUSIONS .....	59
<b>5.</b>	<b>GLOSSARI .....</b>	<b>60</b>
<b>6.</b>	<b>BIBLIOGRAFIA .....</b>	<b>62</b>

## Índex de figures

Fig. 1.	Cicle de vida .....	8
Fig. 2.	Diagrama de Gantt.....	11
Fig. 3.	Productes obtinguts .....	12
Fig. 4.	Diagrama de cas d'ús: visió general .....	16
Fig. 5.	Perfils d'autorització .....	18
Fig. 6.	Menús dels usuaris (web) .....	19
Fig. 7.	Diagrama de cas d'ús: funcionalitats del client web .....	23
Fig. 8.	Pantalles web: benvinguda i autenticació .....	30
Fig. 9.	Pantalles web: agenda.....	30
Fig. 10.	Pantalles web: detall de l'agenda.....	31
Fig. 11.	Pantalles web: perfil personal .....	31
Fig. 12.	Pantalles web: manteniment d'usuari.....	32
Fig. 13.	Diagrama de cas d'ús: funcionalitats del client Android .....	36
Fig. 14.	Pantalles Android: benvinguda i autenticació.....	40
Fig. 15.	Pantalles Android: agenda (format tauleta).....	40
Fig. 16.	Pantalles Android: agenda (2) (format tauleta) .....	41
Fig. 17.	Pantalles Android: agenda (format mòbil/vertical).....	41

Fig. 18. Arquitectura del projecte .....	43
Fig. 19. Diagrama d'entitat-relació .....	46
Fig. 20. Recursos del projecte Web .....	48
Fig. 21. Estructura de paquets del projecte Web .....	48
Fig. 22. Diagrama de classes d'entitat .....	49
Fig. 23. Diagrama de classes DAO .....	50
Fig. 24. Implementació d'Struts2 .....	51
Fig. 25. Patró MVC .....	52
Fig. 26. Esquema JSP i Tiles .....	53
Fig. 27. Recursos del projecte Android .....	55
Fig. 28. Diagrama de paquets del projecte Android .....	55

## Índex de taules

Taula 1. Classificació dels objectius .....	7
Taula 2. Tasques del projecte .....	9
Taula 3. Tasques: dates d'entrega .....	10
Taula 4. Prevenició dels riscos .....	15
Taula 5. Correcció dels riscos .....	15
Taula 6. Camps de l'agenda .....	20
Taula 7. Camps del perfil de l'usuari .....	21
Taula 8. Camps de l'edició de l'usuari .....	22
Taula 9. Camps de l'edició del client .....	22
Taula 10. Camps de l'edició del contacte .....	22
Taula 11. Cas d'ús: autenticació de l'usuari .....	24
Taula 12. Cas d'ús: control de sessió de l'usuari .....	25
Taula 13. Cas d'ús: Accés des d'una aplicació externa independent .....	25
Taula 14. Cas d'ús: creació d'usuari .....	26
Taula 15. Cas d'ús: modificació d'usuari .....	27
Taula 16. Cas d'ús: esborrat d'usuari .....	28
Taula 17. Cas d'ús: modificació del perfil personal/clau .....	28
Taula 18. Cas d'ús: creació/modificació d'esdeveniment .....	29
Taula 19. Cas d'ús: autenticació de l'usuari (Android) .....	37
Taula 20. Cas d'ús: control de sessió de l'usuari (Android) .....	38
Taula 21. Cas d'ús: modificació d'esdeveniment (Android) .....	39
Taula 22. Cas d'ús: refrescar les dades de l'agenda (Android) .....	39
Taula 23. Valoració econòmica .....	58

# 1. Introducció

El propòsit de MobiDroid és proporcionar als usuaris d'una empresa una **agenda col·laborativa, senzilla i interactiva**, on consultar la informació rellevant dels seus esdeveniments, com la ubicació, usuaris assignats, client, contactes. Els usuaris, a més, podran interaccionar amb la informació, de mode que podran canviar el seu estat o posar comentaris visibles per a la resta d'usuaris que també hi estiguin assignats.

Juntament amb aquesta eina, tindran al seu abast una web de gestió per a poder dur un **seguiment global i administrar tant l'agenda com les dades mestres** necessàries per al seu funcionament.

Comentar que **MobiDroid és una marca fictícia**. En el moment de registrar el nom s'hauria de revisar si ja existeix i en cas necessari, es buscaria un nom vàlid.

En els següents punts d'aquesta introducció comentaré els objectius del Treball Fi de Carrera (TFC), com s'ha dut a terme, la planificació i desviaments produïts, i una breu descripció dels productes desenvolupats.

## 1.1. Justificació del TFC

Haver escollit el desenvolupament d'una aplicació J2EE com a temàtica del TFC té com a motiu principal aprendre el màxim possible d'aquest paradigma en el que posaré en pràctica molts conceptes estudiats durant la carrera d'ETIG, i en un projecte que podria arribar a ser aplicat a la realitat.

Disposar dels coneixements en programació orientada a objectes abans d'iniciar el TFC és una condició necessària però no suficient per a poder desenvolupar-lo. Per tant, aquest és el meu punt de partida, tot lo après fins ara que m'ajudarà a assimilar i fer ús d'altres tecnologies en poc temps, com bé podria passar en l'entorn laboral, on l'informàtica canvia constantment.

Per altra banda, aquest TFC complementa i engreixa els meus estudis i, havent-me donat la llibertat d'escollir Android com a client, m'aporta un plus de motivació de cara a iniciar altres projectes semblants al finalitzar l'enginyeria tècnica informàtica.

## 1.2. Objectius del TFC

Els objectius que m'he proposat per al TFC són els següents:

- **Obj1.** Assolir els coneixements tècnics de la **plataforma J2EE** necessaris per a la realització d'aquest projecte.
- **Obj2. Seguir la planificació** establerta, arribant a temps a totes les PACs i, sobretot, al lliurament final.
- **Obj3.** Desenvolupar el producte seguint el **model MVC**, recolzant-me en Struts2.
- **Obj4.** El producte ha de ser fàcilment **portable**. L'ús de l'últim SDK d'Android ha de permetre fer ús de l'aplicació tant a mòbils com a tauletes, sense haver de descartar funcionalitats segons el dispositiu utilitzat sempre i quan la versió del sistema operatiu Android sigui la 4.0.3 o superior. Per a la part web, s'intentarà que l'experiència de l'usuari sigui l'adequada als navegadors web més habituals.
- **Obj5.** A més, el producte ha de ser **modular i escalable**, que permeti en un futur o fins i tot durant la realització del projecte, la incorporació d'altres funcionalitats desitjades que en principi queden fora de l'abast del TFC.
- **Obj6.** Aconseguir un **entorn amigable i intuïtiu** per a l'usuari, intentant minimitzar els clics en el client Android i posant en pràctica mecanismes Ajax al client web.
- **Obj7. Complir amb els estàndards**, en particular el i18n, mostrant als clients Android i web els llenguatges: català, castellà i anglès.
- **Obj8.** Preparar i realitzar una fase de proves amb l'objectiu de **minimitzar els errors** en els productes a lliurar.
- **Obj9.** Utilitzar **programari lliure**
- **Obj10. Referenciar les fonts** utilitzades.

### 1.2.1. Classificació dels objectius

Objectiu	Crític	Prioritari	Secundari
Obj1. Plataforma J2EE	SI		
Obj2. Seguir la planificació	SI		
Obj3. Model MVC	SI		
Obj4. Portable		SI	
Obj5. Modular i escalable		SI	
Obj6. Entorn amigable i intuïtiu			SI
Obj7. Complir els estàndards	SI		
Obj8. Minimitzar els errors			SI
Obj9. Programari lliure		SI	
Obj10. Referenciar les fonts		SI	

Taula 1. Classificació dels objectius



### 1.3. Enfocament i mètode seguit

La realització d'aquest TFC s'ha dut a terme tenint en compte des del primer moment l'alt grau de dificultat per tots els conceptes nous que havia d'aprendre i també les escasses setmanes de desenvolupament disponibles. Per tant, hi havia un risc evident de no poder completar totes les funcionalitats o fins i tot, de quedar-me a mig camí.

De totes maneres, he preferit marcar-me uns objectius indispensables, comentats al punt anterior, i d'altres no tan prioritaris, i durant la vida del projecte revisar-los i en tot cas descartar aquelles funcionalitats que no aportessin gran valor. A les conclusions d'aquest TFC entraré en més detall sobre el resultat obtingut.

Pel que fa al mètode seguit, he tractat de seguir el cicle de vida clàssic tenint en compte que la temporització del projecte no permetia moltes revisions. Tot i això, una vegada realitzat l'anàlisi funcional i tècnic, la programació inicial s'ha centrat en tancar aquells punts imprescindibles que donessin prou garanties de poder entregar el projecte, i a partir d'aquí, anar tancant les funcionalitats extres, revisar el *look and feel* i corregir les incidències que anessin apareixent.

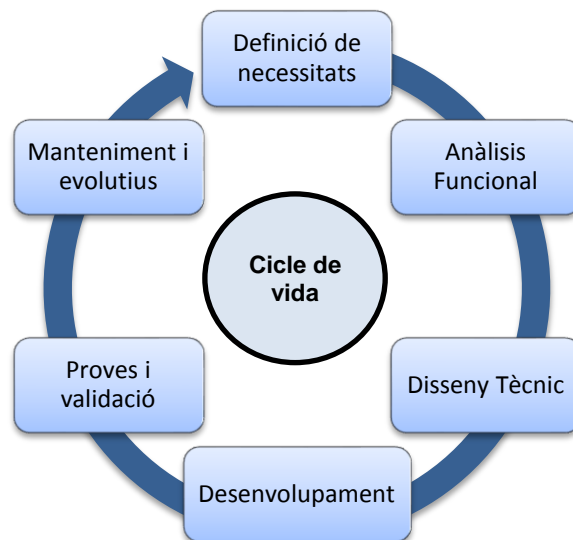


Fig. 1. Cicle de vida

Per últim, comentar que una de les claus era un seguiment el més fidel possible de la planificació, i per tant un dels grans objectius era arribar a cada entrega de les PAC. A les conclusions també comentaré com ha funcionat la planificació.

## 1.4. Planificació del projecte

El TFC contempla les següents tasques:

Tasca	Subtasca	
<b>Pla de Treball (PAC 1)</b>	Presa de requeriments	
	Planificació detallada	
<b>Entrega de la PAC1</b>		
<b>Disseny de la solució (PAC 2)</b>	Proves dels frameworks	
	Anàlisi funcional	
	Disseny tècnic OO	
	Disseny de la base de dades	
	Disseny del prototipus	
	Preparació de l'entorn de desenvolupament	
	Entrega de les proves realitzades	
	<b>Entrega de la PAC2</b>	
	<b>Desenvolupament de la solució (PAC 3)</b>	Modelat de la base de dades
Desenvolupament del servidor: bean, dao		
Desenvolupament client Android-servidor		
Proves client Android-servidor		
Correccions client Android-servidor		
Desenvolupament client web-servidor		
Proves client web-servidor		
Correccions client web-servidor		
Proves integrades Android i web		
Correccions clients Android i web		
Millores en la interfície gràfica		
<b>Entrega de la PAC3</b>		
<b>Lliurament final</b>		Redacció de la memòria
	Revisió de la memòria	
	Creació de la presentació	
	Revisió de la presentació	
<b>Lliurament final</b>		

Taula 2. Tasques del projecte

Les dates previstes d'entrega de cada subtasca són les indicades en el calendari següent.

Subtasca	Data inici	Data final	Duració (d)
Presa de requeriments	02/03/2012	10/03/2012	9
Planificació detallada	11/03/2012	13/03/2012	3
<b>Entrega de la PAC1</b>	14/03/2012	<b>14/03/2012</b>	1
Total PAC1	02/03/2012	14/03/2012	13
Proves dels frameworks	15/03/2012	23/03/2012	9
Anàlisi funcional	24/03/2012	01/04/2012	9
Disseny tècnic OO	02/04/2012	08/04/2012	7
Disseny de la base de dades	09/04/2012	10/04/2012	2
Disseny del prototipus	11/04/2012	11/04/2012	1
Preparació de l'entorn de desenvolupament	12/04/2012	18/04/2012	7
Entrega de les proves realitzades	19/04/2012	19/04/2012	1
<b>Entrega de la PAC2</b>	19/04/2012	<b>19/04/2012</b>	1
Total PAC2	15/03/2012	19/04/2012	36
Modelat de la base de dades	20/04/2012	21/04/2012	2
Desenvolupament del servidor: bean, dao	22/04/2012	25/04/2012	4
Desenvolupament client Android-servidor	26/04/2012	07/05/2012	12
Proves client Android-servidor	08/05/2012	09/05/2012	2
Correccions client Android-servidor	10/05/2012	15/05/2012	6
Desenvolupament client web-servidor	16/05/2012	19/05/2012	4
Proves client web-servidor	20/05/2012	21/05/2012	2
Correccions client web-servidor	22/05/2012	25/05/2012	4
Proves integrades Android i web	26/05/2012	27/05/2012	2
Correccions clients Android i web	28/05/2012	29/05/2012	2
Millores en la interfície gràfica	30/05/2012	03/06/2012	5
<b>Entrega de la PAC3</b>	04/06/2012	<b>04/06/2012</b>	1
Total PAC3	20/04/2012	04/06/2012	46
Redacció de la memòria	05/06/2012	12/06/2012	8
Revisió de la memòria	13/06/2012	14/06/2012	2
Creació de la presentació	15/06/2012	17/06/2012	3
Revisió de la presentació	18/06/2012	18/06/2012	1
<b>Lliurament final</b>	18/06/2012	<b>18/06/2012</b>	1
Total Lliurament final	05/06/2012	18/06/2012	14
<b>Total TFC</b>	<b>02/03/2012</b>	<b>18/06/2012</b>	<b>109</b>

Taula 3. Tasques: dates d'entrega

En el diagrama de Gantt es pot visualitzar la temporització del projecte:

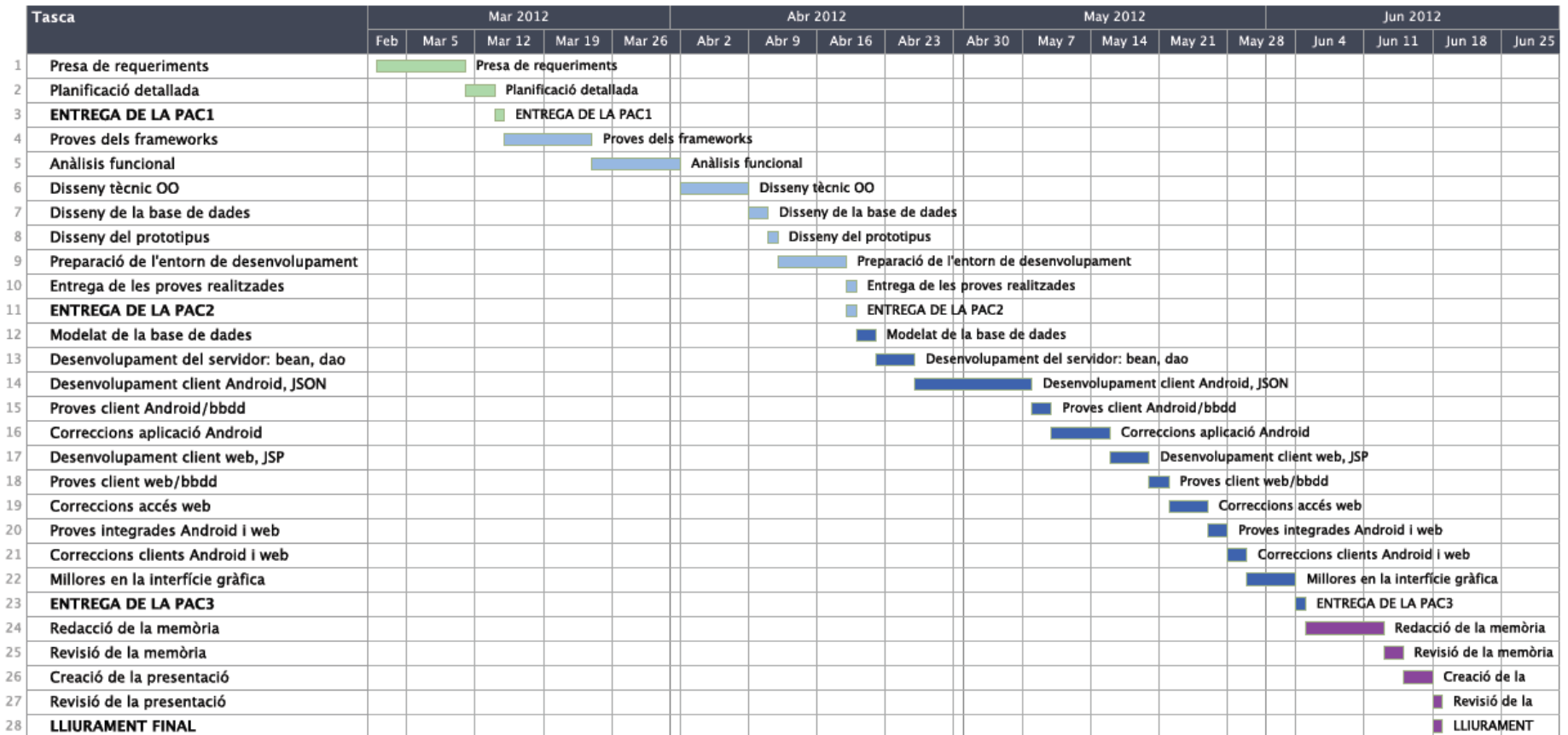


Fig. 2. Diagrama de Gantt

## 1.5. Productes obtinguts

El projecte es compon de següents productes principals:

- **Servidor MobiDroid.** Contindrà la capa controladora amb els DAO per als accessos a la base de dades MySQL mitjançant Hibernate, capa model amb les classes entitat, i capa vista per a la generació de les pàgines a mostrar al client web.
  - **Client Web.** Es mostraran les pàgines JSP generades al servidor. En aquest client es realitzarà el seguiment global de l'agenda i el manteniment dels usuaris i clients de l'aplicació.
- **Client Android.** Aplicació per a la consulta online de l'agenda d'un usuari. Podrà interactuar amb la informació disposada, podent canviar l'estat d'un esdeveniment i desar comentaris que seran compartits amb la resta d'usuaris que també hi estan assignats. Altres funcionalitats seran la possibilitat de realitzar trucades o enviar correus electrònics a aquests usuaris i als contactes del client, en cas que n'hi hagi a l'esdeveniment, i la navegació a la ubicació designada mitjançant Google Maps o Google Navigator.

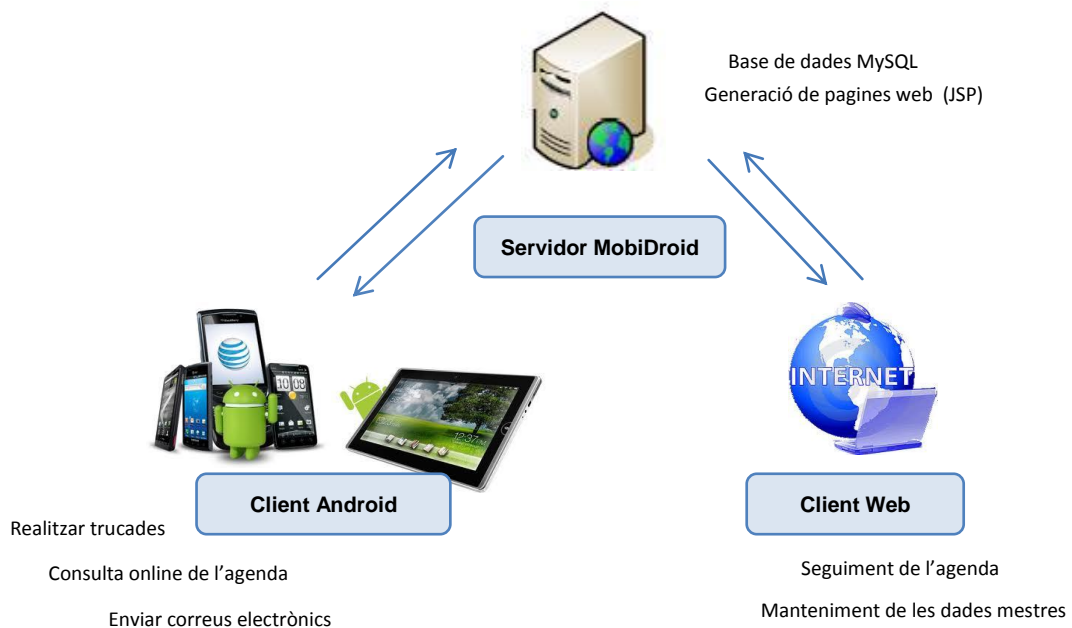


Fig. 3. Productes obtinguts

S'entregaran els fitxers següents, que es poden descarregar des de:

[https://dl.dropbox.com/u/83339852/jperezgomez0/JordiPerez\\_TFC\\_J2EE.zip](https://dl.dropbox.com/u/83339852/jperezgomez0/JordiPerez_TFC_J2EE.zip)

- Codi font projecte Web, amb JavaDoc generat, i compilat MobiDroid.war
- Codi font projecte Android, amb JavaDoc generat, i compilat MobiDroidApp.apk
- *Script* mobidroid.sql per a la creació de la base de dades i joc de proves

## **1.6. Contingut dels altres punts de la memòria**

El següent apartat mostrarà la solució tècnica i funcional del projecte.

Per una banda, es descriuran les funcionalitats del conjunt de productes, tant les desenvolupades com les pendents, així com explicar les millores que es podrien arribar a realitzar.

A continuació, es detallarà la composició tècnica, explicant breument en què consisteix cada framework i llenguatge de programació, i intentant raonar com i perquè s'ha fet servir. Enllaçant amb aquests punts, mostraré els dissenys tècnics, que ajudaran a comprendre la implementació de tot el conjunt.

A la resta d'apartats, mostraré la valoració econòmica del producte, tenint en compte la dedicació en hores que he realitzat, i exposaré les meves conclusions una vegada finalitzat el projecte.

## 2. Presa de requeriments

### 2.1. Restriccions

- Es desenvoluparà fent servir únicament programari lliure.
- L'aplicació Android seguirà els seus estàndards de disseny.
- El client web haurà de ser compatible i seguir els estàndards dels navegadors més usats: IE Explorer, Firefox, Chrome.
- La data límit del projecte és el 18 de juny de 2012.

### 2.2. Normativa i legislació

L'aplicació ha de ser construïda complint els paràmetres establerts per la LOPD (Llei Orgànica de Protecció de Dades de Caràcter Personal):

<http://www.agpd.es/portalwebAGPD/index-ides-idphp.php>

### 2.3. Avaluació de riscos

La següent llista conté els riscos del projecte a desenvolupar:

- **Risc1. Errors en seguretat**, permetent accessos no desitjats a la informació privada dels usuaris o clients.
- **Risc2. Fallides del servidor/s**, provocarien l'aturada del conjunt de l'aplicació.
- **Risc3. Planificació mal temporitzada**, provocant endarreriments, exclusions de funcionalitats previstes o fins i tot no poder entregar el projecte.
- **Risc4. Excés de funcionalitats** previstes, derivat del risc previ, i que pot causar la pèrdua de motivació al veure que no s'hi arriba a temps a totes.
- **Risc5. Insuficiència de funcionalitats**, que pot donar la sensació d'entregar un producte poc treballat i amb uns objectius poc definits.
- **Risc6. Insuficiència de proves** realitzades, amb la pèrdua de qualitat que es produirà a l'aplicació al tenir una major probabilitat d'errors.

## 2.4. Pla de contingència

Risc	Prevenció
Risc1. Errors en seguretat	Base de dades protegida per contrasenya, accés dels usuaris mitjançant claus privades, i control de sessions al servidor. Altres mesures com firewalls físics i lògics.
Risc2. Fallides del servidor	Disposar de varis servidors que permetin un balanceig eficient. Còpies de seguretat periòdiques de les dades.
Risc3. Planificació mal temporitzada	Ser coherent a l'hora de fixar les dates d'entrega i a l'hora d'establir les prioritats de les funcionalitats a desenvolupar.
Risc4. Excés de funcionalitats	Preparar un disseny modular, que permeti diferenciar les funcionalitats, per a poder establir prioritats i dictaminar les indispensables per a l'entrega final.
Risc5. Insuficiència de funcionalitats	S'ha de treballar per garantir un mínim de funcionalitats que donin sentit a la definició del producte.
Risc6. Insuficiència de proves	Preparar la fase de proves i establir un marge per a les correccions.

Taula 4. Prevenció dels riscos

Risc	Correcció
Risc1. Errors en seguretat	Detectar el gap que ha permès l'accés no autoritzat.
Risc2. Fallides del servidor	Revisió de l'errada, càrrega de backups en cas necessari.
Risc3. Planificació mal temporitzada	Nova revisió de la planificació i funcionalitats a entregar. Estimar si serà viable l'entrega del projecte en la data fixada.
Risc4. Excés de funcionalitats	Posposar les prioritats secundàries, estimar si s'ha d'augmentar l'esforç o si s'han de descartar aquestes funcionalitats.
Risc5. Insuficiència de funcionalitats	Donar una empenta a les funcionalitats més importants, intentar afegir-ne de noves. En última instància, completar la memòria detallant les funcionalitats no implementades però necessàries per a que el producte no perdi la seva finalitat.
Risc6. Insuficiència de proves	Revisar els errors produïts, l'impacte d'aquests, i decidir la viabilitat de la correcció per a la data d'entrega. En cas necessari, decidir si la funcionalitat afectada s'haurà de posposar fins a assegurar el seu correcte funcionament.

Taula 5. Correcció dels riscos



### 3. Especificacions funcionals

#### 3.1. Visió general

Com ja s'ha explicat, MobiDroid té dues aplicacions diferenciades:

- **Client web**, on es realitza la gestió i seguiment de l'agenda i el manteniment d'usuaris, clients i contactes d'aquests.
- **Client Android**, per a consultar l'agenda personal, amb la possibilitat d'interactuar amb la informació proporcionada.

Hem de tenir en compte que l'agenda té un caire laboral, i basant-me amb aquest enfocament diferenciarem tres tipus d'usuaris. Ens trobarem amb els caps de departament o supervisors, que a MobiDroid seran els **usuaris gestors** i, per tant, són els qui gestionaran l'agenda del seu personal. Per sota, ens trobarem amb els **usuaris bàsics**, que revisaran els esdeveniments assignats, amb l'opció d'actualitzar-ne l'estat i posar comentaris per a que els seus caps i companys estiguin informats en tot moment de la seva situació. Per últim, ens podem trobar el cas que un gestor vulgui delegar responsabilitat en algun dels seus supervisats i li assigni el rol d'**usuari avançat** que també tindrà la possibilitat de gestionar l'agenda.

Al client Android les funcions són les mateixes independentment del perfil de l'usuari .

En el diagrama de cas d'ús següent es poden visualitzar els principals actors al conjunt de l'aplicació. En aquest primer diagrama, tot i que els usuaris seran els mateixos, diferenciarem els actors Web i Android:

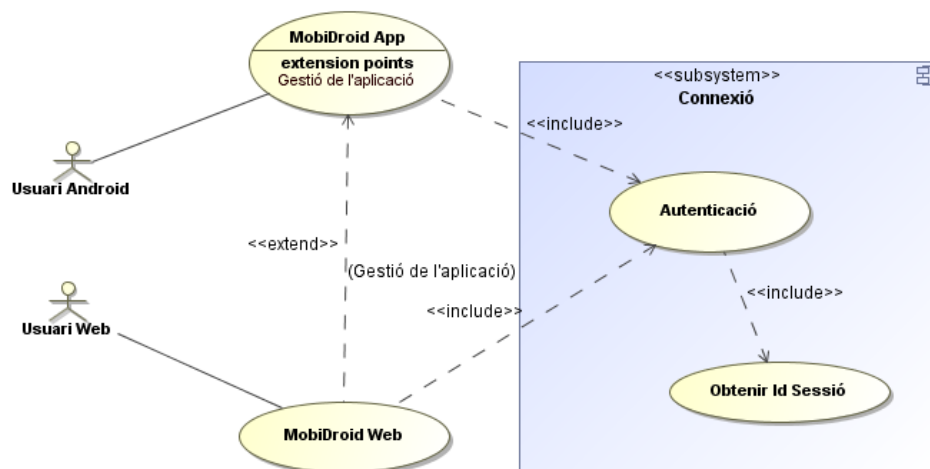


Fig. 4. Diagrama de cas d'ús: visió general

Del diagrama es pot deduir que els dos clients tenen en comú l'**autenticació de l'usuari**, que es realitza de la mateixa manera. Una vegada autenticats, els usuaris obtenen l'**ID de Sessió**, que serà necessari per a l'accés al servidor en la resta de crides. El sistema verificarà que aquest ID de Sessió sigui el vàlid, l'últim generat per l'usuari, i en cas que no sigui així rebutjarà la petició.

En el següents apartats es comenten amb detall les diferents funcionalitats de MobiDroid sense fer referències a aspectes tècnics. Tant en el client web com en el client Android, es dividiran aquests detalls en els següents punts:

- Objectiu
- Perfils d'autorització
- Funcionalitats del client web
- Casos d'ús
- Prototipus de pantalles
- Propostes de millora

## 3.2. Gestió de MobiDroid. El client web

### 3.2.1. Objectius i requisits

Com ja s'ha explicat, **el propòsit del client web és la gestió de l'agenda i de les dades mestres**. Tot i que òbviament l'administrador de sistemes de l'empresa ha de poder accedir en aquesta web, la simplicitat de les dades a administrar dona peu a que sigui el propi supervisor o cap d'equip qui s'encarregui de dur el seu manteniment. En l'apartat "Propostes de millora" es tractaran una sèrie de recomanacions que poden ajudar a optimitzar aquest manteniment.

Per altra banda, haurà de complir una sèrie de característiques:

- Garantir la **seguretat de les dades**, amb un accés protegit per contrasenyes i amb un control eficient de la sessió.
- **Multi llenguatge**, seguint l'estàndard **i18n** i acceptant al menys: català, castellà i anglès. S'haurà de poder escollir l'idioma des de la pantalla d'accés.
- **Modular i escalable**, que ens permeti continuar evolucionant qualsevol apartat sense que la resta es vegi afectat.

- **Amigable i intuïtiu**, intentant que la navegació als diferents apartats no requereixi llegir un text d'ajuda i que les opcions siguin clares i precises.

### 3.2.2. Perfils d'autorització

A l'apartat 3.1 *Visió general* ja hem introduït els diferents tipus d'usuaris existents a MobiDroid. Resumim aquests perfils d'autorització en:

- **Usuari bàsic**: pot veure la seva agenda, sense opció a modificar-la, i pot accedir al seu perfil d'usuari per a canviar telèfon, correu electrònic o la clau personal d'accés.
- **Usuari avançat**: pot actualitzar el seu perfil d'usuari tal i com ho fa l'usuari bàsic i a més pot modificar l'agenda.
- **Usuari gestor**: les mateixes opcions que l'usuari avançat, però també té accés al manteniment d'usuaris i al de clients i contactes.

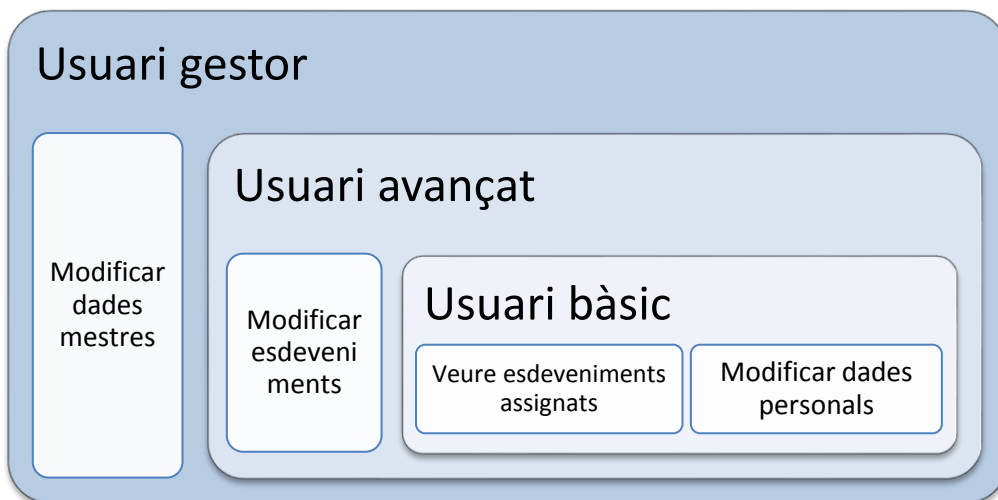


Fig. 5. Perfils d'autorització

Possiblement s'hauria pogut simplificar el quadre deixant únicament dos perfils d'usuaris, però s'ha considerat apropiat deixar un mínim de 3 i que sigui l'empresa qui decideixi si vol fer-ne servir únicament dos (inutilitzant o l'usuari avançat o el bàsic).

De totes maneres, es recomana fer servir de bon inici una assignació lògica de perfils d'autorització tot i que les diferències no siguin molt destacades, ja que de cara a futures ampliacions del programari haver de **refer el quadre d'autoritzacions pot ser una tasca costosa i delicada** si la quantitat d'usuaris és molt gran.

Per últim, la figura de l'administrador és pot veure reflectida en el gestor, per la característica de tenir accés al manteniment de les dades mestres de l'aplicació.

### 3.2.3. Funcionalitats del client web

Les funcionalitats principals del client web són les següents:

- Menú personalitzat per a cada perfil d'usuari
- Agenda personal
- Gestió de l'agenda
- Perfil de l'usuari
- Manteniment d'usuaris
- Manteniment de clients i contactes

#### 3.2.3.1. Menú personalitzat per a cada perfil d'usuari

Els menús seran diferents depenent del perfil de l'usuari:



Fig. 6. Menús dels usuaris (web)

### 3.2.3.2. Agenda Personal

Els usuaris bàsics tindran l'opció de visualitzar l'agenda personal des de la web.

Tot i que aquesta opció també és disponible en l'aplicació per a mòbils i tauletes, s'ha decidit incloure-la en la versió web ja que el cost de la implementació era mínim al disposar de l'opció Gestió de l'Agenda que permet visualitzar l'agenda de tots els usuaris. A més, d'aquesta manera es garanteix que un usuari pugui tenir accés a aquesta informació des de qualsevol dispositiu amb navegador i no limitar-lo a una aplicació per Android.

Tot i això, cal remarcar que **les funcionalitats extres que gaudeix la versió Android van més enllà de la simple visualització de l'agenda**, tal i com es detallarà a l'apartat corresponent.

### 3.2.3.3. Gestió de l'Agenda

Des d'aquesta pantalla **els usuaris avançats i gestors podran dur el seguiment de tota l'agenda**, amb l'opció de filtrar per data d'inici, estat, títol, usuaris assignats i client.

Des de la llista es podrà accedir al detall d'un esdeveniment fent clic al seu títol. Aquesta pantalla de detall serà la mateixa que es mostra al crear un nou esdeveniment. La informació que es mostrarà és la següent:

Camp	Domini
Títol	
Categoria	tasca, visita, reunió, entrega...
Estat	pendent, en curs, finalitzat, posposat, cancel·lat, esborrat
Data i hora d'inici	
Data i hora fi	
Client	clients donats d'alta al sistema
País	
Localitat	
Direcció	
Escala, pis, porta..	
Descripció	
Usuaris assignats	usuaris donats d'alta al sistema

Taula 6. Camps de l'agenda

### 3.2.3.4. Perfil de l'usuari

Amb aquesta opció, l'usuari accedirà a les seves dades personals, però **només podrà modificar el telèfon, el correu electrònic i la seva clau d'accés.**

Els camps que visualitzarà són aquests:

Camp
Nif/DNI
Nom
Cognoms
Departament
Telèfon
Mail
Clau antiga
Clau nova
Confirmació de clau

Taula 7. Camps del perfil de l'usuari

### 3.2.3.5. Manteniment d'usuaris

Al manteniment d'usuaris, **els gestors podran crear, modificar, esborrar o donar de baixa usuaris.**

Des del llistat es podrà accedir a l'edició de l'usuari fent clic al Nif/DNI de l'usuari i també podrà esborrar-lo seleccionant aquesta opció que apareix a l'última columna del llistat. **Per tal de mantenir la integritat de la base de dades, s'han pres mesures de seguretat al esborrar un usuari.** En el cas que el sistema detecti que l'usuari ha accedit en al menys una ocasió o estigui assignat en algun esdeveniment, no es permetrà esborrar l'usuari sinó que quedarà marcat com a inactiu (sempre serem a temps d'activar-lo de nou).

**Al crear un nou usuari, s'establirà una clau per defecte, composta pels últims 4 dígit del NIF informat.** A més, aquesta clau podrà ser restaurada al editar un contacte, sense necessitat de conèixer la clau anterior.

A remarcar que, **per raons de seguretat, el gestor no es podrà esborrar a si mateix ni modificar el seu perfil d'autorització** (és a dir, no pot decidir deixar de ser gestor).

Els camps que es visualitzaran al editar o crear els usuaris són els següents:

Camp
Nif/DNI
Nom
Cognoms
Perfil
Departament
Telèfon
Mail
Actiu
Clau nova
Confirmació de clau

Taula 8. Camps de l'edició de l'usuari

### 3.2.3.6. Manteniment de clients i contactes

Aquest manteniment serà molt semblant al d'usuaris, podent **crear, editar o esborrar clients**. Els camps d'un client són aquests:

Camp
Cif
Nom
Contactes

Taula 9. Camps de l'edició del client

Com en el manteniment d'usuaris, **el sistema no permetrà esborrar un client si està assignat a un esdeveniment**.

Al accedir a l'edició d'un client, es visualitzarà la llista de contactes, on també es podran crear, editar o esborrar. Dels contactes disposarem de la següent informació:

Camp
Nom del contacte
Telèfon
Mail

Taula 10. Camps de l'edició del contacte

### 3.2.4. Casos d'ús

Els **actors** del client del client web seran els diferents tipus d'usuaris: bàsic, avançat, gestor. Es separaran els casos d'ús per subsistemes: connexió, manteniment i agenda.

En els casos d'ús no es detallarà el control dels errors no desitjats, com seria el cas d'un error a la base de dades. Tot i això, l'aplicació gestionarà aquests errors i mostrarà els missatges pertinents a l'usuari. Per tant, les excepcions que es detallaran fan referència a les validacions de negoci.

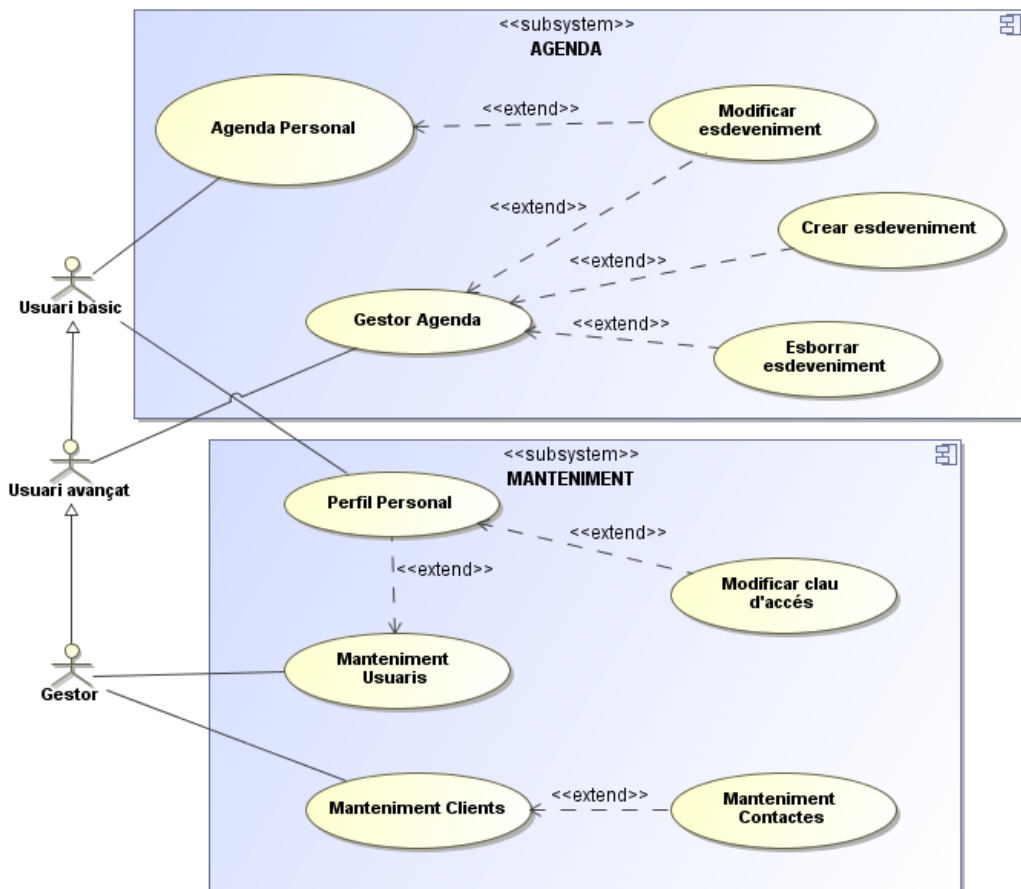


Fig. 7. Diagrama de cas d'ús: funcionalitats del client web



### 3.2.4.1. Connexió

Els casos d'ús de connexió detallaran com s'accedeix a la web i com es controla la sessió una vegada l'usuari ja està autenticat. Aquest segon cas d'ús serà present a la resta de casos d'ús, ja que el control es realitza durant tota la sessió de l'usuari.

Cas d'ús	1.1 Autenticació de l'usuari
Descripció	Accés de l'usuari a l'aplicació i autenticació al sistema per a obtenir l'ID de sessió
Actors	Tots (bàsic, avançat, gestor)
Casos d'us relacionats	
Precondició	L'usuari ha accedit a la web de gestió de MobiDroid
Condicció de fi amb èxit	El servidor autoritza l'ús de l'aplicació al client, enviant l'ID de sessió
Condicció de fi amb fracàs	El servidor NO autoritza l'ús de l'aplicació al client, mostrant un missatge d'error
Alternatives de procés	1- L'usuari accedeix a MobiDroid 2- L'usuari informa el NIF i la clau de 4 dígits. 3- El sistema comprova que el NIF de l'usuari existeixi. 4- El sistema comprova que la clau és vàlida. 5- El sistema genera el codi de sessió, que es tracta de la concatenació del dia en format americà (AAAAMMDD) i hora. 6- El sistema desa el codi de sessió a base de dades. 7- El servidor retorna el codi de sessió a l'aplicació client.
Excepcions	3.1- El sistema detecta que l'usuari no existeix i mostra un missatge d'error. 4.1- El sistema detecta que la clau no és vàlida i mostra un missatge d'error.
Altres comentaris	El sistema no comprovarà si l'usuari no ha tancat la sessió amb anterioritat, sempre generarà un nou codi de sessió

Taula 11. Cas d'ús: autenticació de l'usuari

Cas d'ús	1.2 Control de sessió de l'usuari
Descripció	Controla que la sessió de l'usuari estigui activa
Actors	Tots (bàsic, avançat, gestor)
Casos d'us relacionats	1.1 Autenticació de l'usuari
Precondició	L'usuari s'ha autenticat correctament i realitza qualsevol acció (clic a

	altres punts de menú, enllaços o botons de la pantalla).
Condicció de fi amb èxit	Validació transparent per l'usuari, el servidor no interromp l'acció.
Condicció de fi amb fracàs	El servidor NO autoritza l'ús de l'aplicació al client, redirigint-lo a la pàgina d'inici i mostrant un missatge d'error
Alternatives de procés	1- El sistema valida que la sessió continua activa. 2- El sistema valida que l'ID Sessió de l'usuari sigui la més recent.
Excepcions	1.1- El sistema detecta que l'usuari s'ha desconnectat del sistema i el retorna a la pàgina d'inici. 2.1- El sistema detecta que l'ID de sessió ha caducat, ja s'ha generat un de més nou i mostra la pàgina d'inici i missatge d'error.
Altres comentaris	L'excepció 2.1 es produirà sempre que l'usuari es torni a autenticar des d'un altre navegador, ordinador o des del client Android, que també genera ID de sessió.

**Taula 12. Cas d'ús: control de sessió de l'usuari**

Cas d'ús	1.3 Accés des d'una aplicació externa independent
Descripció	Accés de l'usuari a l'aplicació i validació d'ID de sessió. En aquest TFC, l'aplicació Android és la que pot sol·licitar l'accés extern sense autenticació per contrasenya.
Actors	Tots (bàsic, avançat, gestor)
Casos d'us relacionats	
Precondició	L'usuari accedeix a la web des d'una aplicació independent autoritzada
Condicció de fi amb èxit	El servidor autoritza l'ús de l'aplicació al client, mostrant directament l'agenda (no mostra la pàgina d'inici)
Condicció de fi amb fracàs	El servidor NO autoritza l'ús de l'aplicació al client, redirigint-lo a la pàgina d'inici i mostrant un missatge d'error
Alternatives de procés	1- El sistema extern envia l'ID d'usuari i l'ID de sessió. 2- El sistema valida que l'ID de sessió de l'usuari sigui la més recent.
Excepcions	1.1- El sistema detecta que no s'ha informat ID d'usuari o ID de sessió i mostra la pàgina d'inici i missatge d'error. 2.1- El sistema detecta que l'ID de sessió ha caducat, degut a que ja s'ha generat un ID més nou, i mostra la pàgina d'inici i l'error.
Altres comentaris	El sistema no comprovarà si l'usuari no ha tancat la sessió amb anterioritat, sempre generarà un nou codi de sessió

**Taula 13. Cas d'ús: Accés des d'una aplicació externa independent**

### 3.2.4.2. Manteniment

En aquest apartat, només ens centrarem en els casos d'ús del manteniment d'usuaris, ja que els del manteniment de clients i de contactes són molt similars i amb menys validacions.

Cas d'ús	2.1 Creació d'usuari
Descripció	Donar d'alta un usuari nou, indicant l'identificador (DNI), el perfil i els atributs corresponents.
Actors	Gestor
Casos d'us relacionats	1.2 Control de sessió de l'usuari
Precondició	L'usuari s'ha autenticat correctament.
Condicció de fi amb èxit	Missatge amb el resultat de l'operació
Condicció de fi amb fracàs	Missatge informant de l'error
Alternatives de procés	1- L'usuari accedeix al manteniment d'usuaris 2- L'usuari selecciona l'opció crear un usuari. 3- L'usuari completa els atributs de l'usuari i prem desar. 4- L'usuari prem desar dades. 5- El sistema verifica que les dades obligatòries s'han informat i que els formats són els correctes. 6-El sistema estableix una contrasenya inicial de 4 dígits, que són els 4 últims del NIF. 7- El sistema comprova que l'usuari no estigui donat d'alta. 8- El sistema genera la clau de l'usuari, que es tracta dels últims 4 dígits del NIF informat per pantalla. 9- El sistema desa les dades al servidor i retorna un missatge informatiu.
Excepcions	5.1- El sistema detecta errors i retorna el missatge corresponent. 7.1- El sistema comprova que l'usuari ja està donat d'alta i retorna un missatge d'error. 8.1-El sistema no pot obtenir els últims 4 dígits de l'identificador i retorna un missatge d'error.
Altres comentaris	Els camps obligatoris són: nif, nom, cognoms, perfil , telèfon, email El perfil sempre estarà informat ja que es selecciona d'una llista desplegable sense valor en blanc.

Taula 14. Cas d'ús: creació d'usuari

Cas d'ús	2.2 Modificació d'usuari
Descripció	Modificar els atributs d'un usuari
Actors	Gestor
Casos d'us relacionats	1.2 Control de sessió de l'usuari
Precondició	L'usuari s'ha autenticat correctament.
Condicció de fi amb èxit	Missatge amb el resultat de l'operació
Condicció de fi amb fracàs	Missatge informant de l'error
Alternatives de procés	1- L'usuari accedeix al manteniment d'usuaris 2- L'usuari selecciona de la llista, un usuari a modificar clicant sobre el seu NIF. 3- L'usuari modifica els atributs de l'usuari i prem desar. 4- L'usuari prem desar dades. 5- El sistema verifica que les dades obligatòries s'han informat i que els formats són els correctes. 6- El sistema verifica que clau nova i confirmació de clau són les mateixes, en cas que estiguin informades. 7- El sistema desa les dades al servidor i retorna un missatge informatiu.
Excepcions	5.1- El sistema detecta errors i retorna el missatge corresponent. 6.1- El sistema detecta que les claus són diferents i retorna un missatge d'error.
Altres comentaris	Els camps obligatoris a modificar són: nif, nom, cognoms, perfil , telèfon, email. El perfil sempre estarà informat ja que es selecciona d'una llista desplegable sense valor en blanc.

Taula 15. Cas d'ús: modificació d'usuari

Cas d'ús	2.3 Esborrat d'usuari
Descripció	Esborrar un usuari del sistema
Actors	Gestor
Casos d'us relacionats	1.2 Control de sessió de l'usuari
Precondició	L'usuari s'ha autenticat correctament.
Condicció de fi amb èxit	Missatge informant si s'ha esborrat l'usuari o s'ha deixat com a inactiu
Alternatives de procés	1- L'usuari accedeix al manteniment d'usuaris 2- L'usuari selecciona de la llista un usuari a esborrar, fent clic sobre l'enllaç per a executar l'acció. 3- El sistema comprova que el NIF de l'usuari existeixi. 4- El sistema comprova si l'usuari si té esdeveniments pendents o

<p>si ha accedit alguna vegada a l'aplicació.                  4.1 - El sistema marca el flag d'inactiu.                  5- El sistema desa les dades al servidor i retorna un missatge informatiu.</p>
Excepcions

**Taula 16. Cas d'ús: esborrat d'usuari**

Cas d'ús	2.4 Modificació del perfil personal/clau
Descripció	Modificació de les dades personals de l'usuari i de la seva contrasenya
Actors	Tots (bàsic, avançat, gestor)
Casos d'us relacionats	1.2 Control de sessió de l'usuari
Precondició	L'usuari s'ha autenticat correctament.
Condicció de fi amb èxit	Missatge amb el resultat de l'operació
Condicció de fi amb fracàs	Missatge informant de l'error
Alternatives de procés	<p>1- L'usuari selecciona l'opció Perfil Personal. En el cas del gestor, accedirà a la mateixa pantalla al editar el seu usuari des del Manteniment d'Usuaris .</p> <p>2- L'usuari modifica el seu número de telèfon, correu electrònic o clau (informant l'antiga, la nova i la confirmació de la nova).</p> <p>3- L'usuari prem desar dades.</p> <p>4- El sistema valida la validesa de les claus informades.</p> <p>5- El sistema desa les dades al servidor i retorna un missatge informatiu.</p>
Excepcions	<p>4.1.1 - El sistema detecta que la clau antiga està informada però no ho està la clau nova, o la clau antiga i la nova estan informades però no ho està la confirmació de clau, i retorna un missatge d'error.</p> <p>4.1.2 - El sistema detecta que la clau antiga no és vàlida, o que les claus noves i de confirmació són diferents i retorna un missatge d'error.</p>

**Taula 17. Cas d'ús: modificació del perfil personal/clau**

### 3.2.4.3. Agenda

Cas d'ús	3.1 Creació/Modificació d'esdeveniment
Descripció	Donar d'alta o modificar un esdeveniment nou a l'agenda.
Actors	Gestor / Usuari avançat
Casos d'us relacionats	1.2 Control de sessió de l'usuari
Precondició	L'usuari s'ha autenticat correctament.
Condicció de fi amb èxit	Missatge amb el resultat de l'operació
Condicció de fi amb fracàs	Missatge informant de l'error
Alternatives de procés	<ol style="list-style-type: none"><li>1- L'usuari accedeix a la gestió de l'agenda.</li><li>2.a- L'usuari selecciona l'opció crear esdeveniment.</li><li>2.b- L'usuari selecciona l'esdeveniment a modificar fent clic al títol.</li><li>3- L'usuari informa el formulari amb les dades de l'esdeveniment.</li><li>4- L'usuari prem desar dades.</li><li>5- El sistema verifica que les dades obligatòries s'han informat i que els formats són els correctes.</li><li>6- El sistema verifica la coherència de les dates inici i final.</li><li>7- El sistema desa les dades al servidor</li><li>8- El sistema modifica l'indicador d'agenda modificada (marcador de data i hora) als usuaris assignats a l'esdeveniment.</li><li>9- retorna un missatge informatiu.</li></ol>
Excepcions	<ol style="list-style-type: none"><li>5.1- El sistema detecta errors i retorna el missatge corresponent.</li><li>6.1- El sistema detecta que la data-hora fi és inferior a data-hora inici i retorna un missatge d'error.</li><li>6.2- El sistema detecta que la data-hora inicial és igual o inferior a la data-hora actual, i mostra un missatge d'error.</li><li>6.3- El sistema detecta que la data-hora fi no està informada, posa la mateixa data-hora que la inicial i mostra un missatge d'advertència.</li></ol>
Altres comentaris	<p>Els camps obligatoris són: títol, categoria, data-hora inicial, usuaris assignats.</p> <p>El camp Estat per defecte al crear un esdeveniment és Pendent.</p>

Taula 18. Cas d'ús: creació/modificació d'esdeveniment

### 3.2.5. Prototipus de pantalles

#### 3.2.5.1. Benvinguda i autenticació

MobiDroid: Gestor de Mobilitat.

Usuari:

Clau:

Fig. 8. Pantalles web: benvinguda i autenticació

#### 3.2.5.2. Agenda

Els menús seran personalitzats per cada perfil d'Usuari, en aquests prototipus mostrem el menú del gestor que és el que té tots totes les opcions.

MobiDroid: Gestor de Mobilitat.

Agenda

Perfil personal

Manteniment usuaris

Manteniment clients

Data inici: 02/07/2012

Data fi: 05/07/2012

Estat:  Títol:  Usuaris assignats:  Empresa:

Títol	Usuaris	Categori	Data inici	Estat	Descripció	Empres	Ubicació
Reurió SIP	Maite Cano	Reurió	02-JU-2012	Penden	Reurió de direcció	SIP.S.A	Sala 1
Visita INS	Joan Font	Visita	02-JU-2012	Penden	Visita amb Ana	INS.S.A	C/Muntaner
Abs.autoritzad	Laura Martinez	Personal	03-JU-2012	Penden	Metge		

Fig. 9. Pantalles web: agenda

### 3.2.5.3. Detall de l'Agenda (crear, editar)

MobiDroid: Gestor de Mobilitat

Desconnectar

Agenda

Perfil personal

Manteniment usuaris

Manteniment clients

Títol: Visita

Estat: Pendent

Data inici: 02-Jul-2012 11:00

Data fi: 02-Jul-2012 12:00

Categoria: Visita

País: Espanya

Població: Barcelona

Adreça: C/Muntaner

Escala, porta, pis: 2 baixos

Descripció: Visita amb Ano

Assigneu usuaris:

Sense assignar

Assignats

Laura Martinez  
Mate Cano

Joan Font

Desar

http://www.mobidroid.cat/agendamod

Fig. 10. Pantalles web: detall de l'agenda

### 3.2.5.4. Perfil Personal

MobiDroid: Gestor de Mobilitat

Desconnectar

Agenda

Perfil personal

Manteniment usuaris

Manteniment clients

NIF/DNI: 47779368-Q

Nom: Laura

Cognoms: Martinez

Departament: Finances

Telèfon \*: 610344562

Mail \*: lmartinez@mail.com

Clau actual:

Clau nova:

Confirmació de clau:

Desar

http://www.mobidroid.cat/perfilmod

Fig. 11. Pantalles web: perfil personal

El camp Perfil quedarà amagat en aquesta pantalla.



### 3.2.5.5. Manteniment d'Usuari

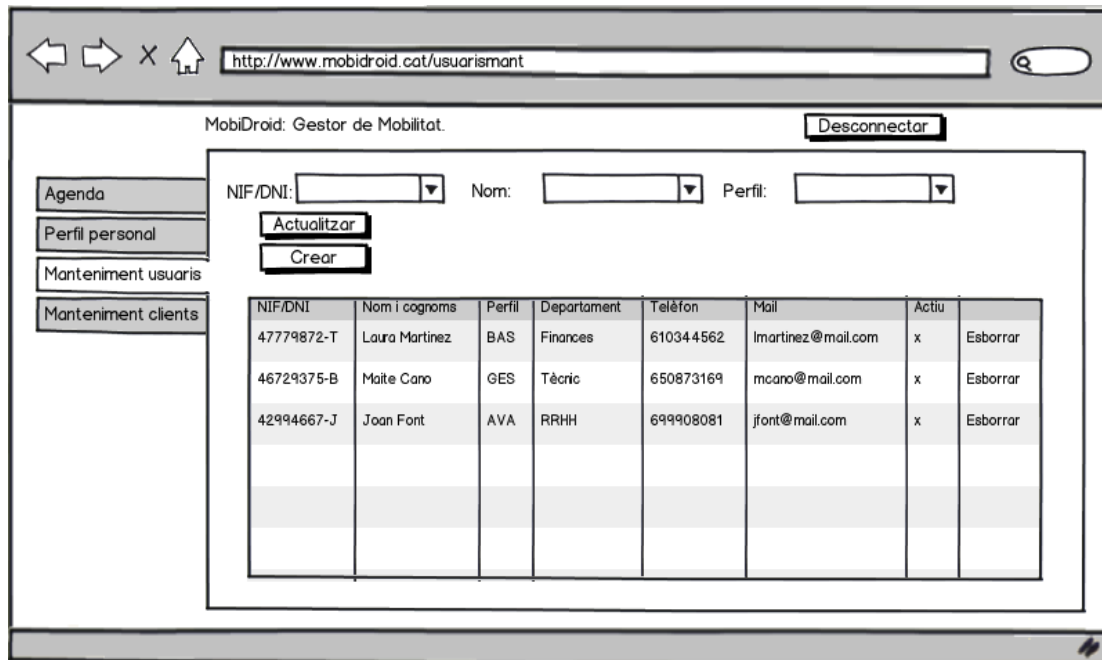


Fig. 12. Pantalles web: manteniment d'usuari

El camp NIF/DNI de la taula contindrà l'enllaç per editar l'Usuari. L'enllaç Esborrar no haurà d'aparèixer en cas que l'Usuari de la taula sigui l'usuari de la sessió.

### 3.2.5.6. Detall de l'Usuari

Mateixa pantalla que Perfil Personal, però mostrant els camps Perfil i Actiu, i amagant Clau actual. El camp Actiu no haurà d'aparèixer en cas que l'Usuari de la taula sigui l'usuari de la sessió.

### 3.2.5.7. Manteniment de Client

Pantalla amb la mateixa estructura i opcions que Manteniment d'Usuari. Els filtres seran Nom i CIF. El camp CIF de la taula contindrà l'enllaç per editar el Client.

### 3.2.5.8. Detall de Client

Mateixa estructura que Perfil Personal, mostrant les dades de Client. També es mostrarà la taula amb els Contactes del Client, on el camp NOM contindrà l'enllaç per a la seva edició. A més, en aquesta taula hi haurà l'opció d'esborrar aquests contactes o crear-ne de nous.

No es mostrarà el codi intern de Client ni el de Contacte.

### 3.2.5.9. Detall de Contacte

Mateixa estructura que Perfil Personal, mostrant les dades de Contacte. No es mostrarà el codi intern del Contacte.

### 3.2.6. Propostes de millora

La gestió de l'agenda té un gran ventall de possibilitats, les enumero:

- Possibilitat de visualitzar l'agenda en format calendari: diari, setmanal, mensual.
- Control de disponibilitat d'usuaris al assignar-los a un esdeveniment. Es mostraria una finestra de confirmació en cas de detectar que un usuari ja està assignat a una altre esdeveniment per a les dates indicades. Actualment, el sistema permet aquest solapament.
- Millorar l'aspecte gràfic.
- Exportar els llistats a Excel, CSV, i fins i tot a PDF.
- Gestió d'alertes per correu electrònic i amb l'opció de missatgeria SMS.
- Interfícies per a la integració de les bases de dades de l'empresa per a alimentar les dades mestres d'usuaris i clients.
- Versió de la web optimitzada per a l'accés des de dispositius mòbils.
- Informes estadístics, que aportin més informació per al seguiment de l'agenda.
- Històric de canvis de les tasques.
- Possibilitat d'afegir diversos clients a un mateix esdeveniment o d'assignar contactes independentment dels clients participants.
- Indicar punt de punt de partida d'un esdeveniment, per a poder calcular quilometratges.

## 3.3. MobiDroid per a dispositius mòbils. El client Android

### 3.3.1. Objectius i requisits

Sóc de l'opinió que una aplicació per a mòbil o tauleta ha de ser senzilla, concisa i molt amigable. A més, ha d'aprofitar al màxim les funcionalitats que ofereixi el sistema operatiu, de cara a aportar valors afegits que una aplicació d'escriptori o web no pot donar. Aquests són els trets principals del producte a desenvolupar, i en base a aquests objectius, es planteja facilitar a l'usuari l'accés a la seva agenda, mostrant la informació necessària i explotant els avantatges que ens proporciona Android.

Per altra banda, s'ha de tenir en compte que el consum de l'accés a Internet des de mòbil és un limitant a dia d'avui, i per tant, un requisit important és que les consultes al servidor no siguin constants i que aquestes siguin eficients i poc pesades.

Per altra banda, l'aplicació haurà de seguir l'estàndard i18n, mostrant els idiomes: català, castellà, anglès. L'idioma que escollirà l'aplicació serà el que l'usuari tingui assignat al Sistema Operatiu del seu mòbil o tauleta.

### **3.3.2. Perfils d'autorització**

Dins l'aplicació no s'apreciaran diferències pel perfil d'usuari, però hi ha d'haver cap problema en incorporar millores que només hagin de mostrar-se per a un tipus en concret.

### **3.3.3. Funcionalitats del client Android**

En el cas de l'aplicació per Android, les funcionalitats principals són aquestes:

- Navegació per les pantalles
- Agenda personal
- Modificacions a l'agenda personal
- Explotació de la informació

#### **3.3.3.1. Navegació per les pantalles**

Es tindrà en compte que l'aplicació es pot visualitzar en horitzontal o vertical, i també la diferència de resolució en cas de l'execució via mòbil o via tauleta.

Per tant, s'ha de garantir que la informació es visualitzarà correctament, i això s'aconseguirà decidint el contingut a mostrar segons els píxels horitzontals. A l'apartat Prototipus de pantalles es donarà més detall.

Una altra característica serà el mètode de transició entre les pantalles de detall de l'agenda. Per a aquesta transició es farà ús de la funcionalitat tàctil dels dispositius mòbils, lliscant la pantalla amb el dit d'esquerra a dreta o al revés.

### 3.3.3.2. Agenda personal

L'usuari visualitzarà la seva agenda personal, mostrant la llista d'esdeveniments per ordre de data d'inici. Per defecte es mostraran els esdeveniments del dia actual i cap en davant, però l'usuari podrà filtrar per una altra data si ho desitja.

Quan esculli l'esdeveniment, es passarà a mostrar el seu detall sense refrescar les dades del servidor. Per tant, **l'aplicació ha d'oferir plena navegació *offline* una vegada ja s'ha autenticat**. En el moment en que sigui necessari connectar-se amb el servidor, si no hi troba connexió ha de mostrar un missatge d'error però no ha de fer fora l'usuari de l'aplicació.

Per a refrescar les dades, pot fer ús del botó Refrescar. És requisit que en una aplicació per a mòbils la **sincronització del client amb el servidor ha de ser òptima**. A MobiDroid, l'ús de marcadors de temps i hora a la base de dades(timestamps) proporcionaran al conjunt de l'aplicació informació suficient per saber si ha d'actualitzar les dades o no i per tant, només es refrescaran les dades si l'agenda de l'usuari si s'han trobat dades noves.

Des de l'agenda personal també es disposarà d'un botó per a navegar a la web de Gestió de l'Agenda, sense necessitat de tornar-se a autenticar.

### 3.3.3.3. Modificacions a l'agenda personal

L'usuari podrà modificar els camps:

- Estat de l'esdeveniment
- Comentaris

Al fer clic sobre l'estat de l'esdeveniment es mostrarà una llista desplegable amb totes les opcions. No hi haurà control dels estats, és a dir, es pot passar d'un estat "Cancel·lat" a un de "Pendent".

Pel que fa als comentaris, es visualitzaran en un quadre de text i ordenats per data d'inserció, de més recent a més antic. Abans del text es mostrarà la data i hora del comentari i l'usuari que l'ha escrit.

Al canviar l'estat o al inserir un comentari es desarà automàticament a la base de dades. A més, el sistema deixarà constància a tots els usuaris assignats que la seva agenda ha canviat, de mode que quan premin el botó Refrescar recuperin aquesta informació.

### 3.3.3.4. Explotació de la informació

Android proporciona eines per a enriquir les aplicacions. En el cas de MobiDroid, s'aprofitaran les següents:

- Enllaç a **Google Maps**, mostrant la ubicació assignada a l'esdeveniment. Android preguntarà si es desitja mostrar a través de la web o utilitzant l'aplicació nativa d'Android.
- Enllaç a **Google Navigator**, permetent navegar fins a la ubicació assignada a l'esdeveniment.
- Opció de **realitzar trucades telefòniques als usuaris assignats i contactes**.
- Opció d'**enviament de correus als usuaris assignats i contacte**.

### 3.3.4. Casos d'ús

A l'aplicació Android tindrem un únic **actor**, que l'anomenem usuari Android.

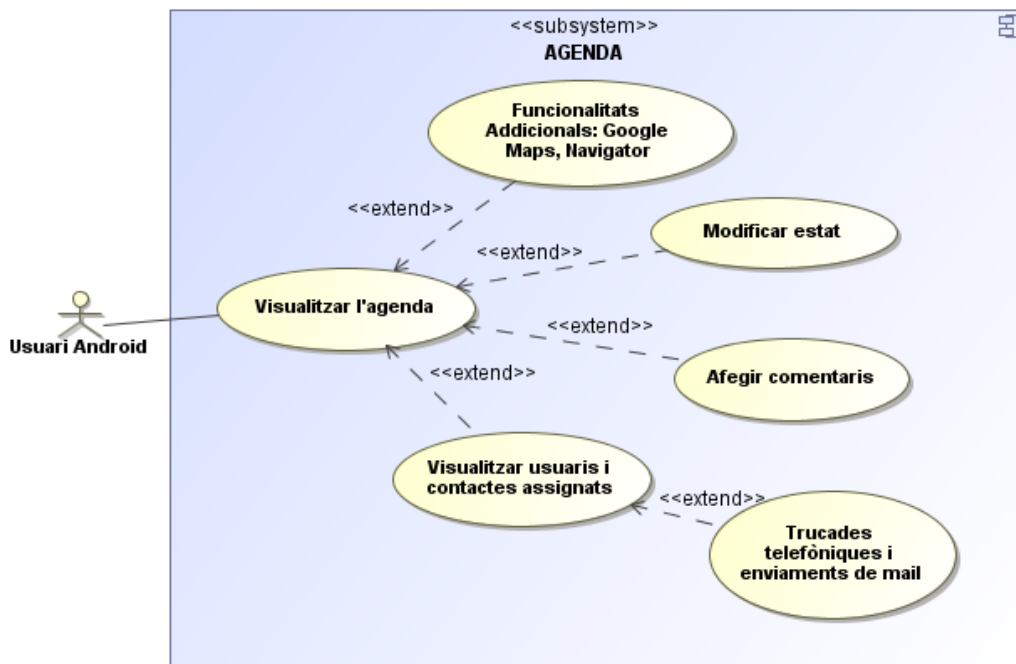


Fig. 13. Diagrama de cas d'ús: funcionalitats del client Android

### 3.3.4.1. Connexió

Cas d'ús	a.1 Autenticació de l'usuari
Descripció	Accés de l'usuari a l'aplicació i autenticació al sistema per a obtenir l'ID de sessió
Actors	Usuari Android
Casos d'us relacionats	
Precondició	L'usuari ha accedit a l'aplicació MobiDroid per a Android
Condicció de fi amb èxit	El servidor autoritza l'ús de l'aplicació al client, enviant l'ID de sessió i les dades de l'agenda.
Condicció de fi amb fracàs	El servidor NO autoritza l'ús de l'aplicació al client, mostrant un missatge d'error. Missatge d'error al no trobar connexió amb el servidor.
Alternatives de procés	1- L'usuari accedeix a MobiDroid 2- L'usuari informa el NIF i la clau de 4 dígits. 3- El sistema comprova que el NIF de l'usuari existeixi. 4- El sistema comprova que la clau és vàlida. 5- El sistema genera el codi de sessió, que es tracta de la concatenació del dia en format americà (AAAAMMDD) i hora. 6- El sistema desa el codi de sessió a base de dades. 7- El servidor retorna el codi de sessió a l'aplicació Android i també totes les dades de l'agenda.
Excepcions	3.1- No hi ha connexió amb el servidor i es mostra un missatge d'error. 3.2- El sistema detecta que l'usuari no existeix i mostra un missatge d'error. 4.1- El sistema detecta que la clau no és vàlida i mostra un missatge d'error.
Altres comentaris	Aquest cas d'ús serà molt similar al "1.1 Autenticació de l'usuari" del client web pel que fa a l'aspecte funcional.

Taula 19. Cas d'ús: autenticació de l'usuari (Android)

Cas d'ús	a.2 Control de sessió de l'usuari
Descripció	Controla que la sessió de l'usuari sigui l'última
Actors	Usuari Android
Casos d'us relacionats	a.1 Autenticació de l'usuari
Precondició	L'usuari s'ha autenticat correctament i intenta:

	<ul style="list-style-type: none"> <li>- Refrescar les dades</li> <li>- Modificar un esdeveniment.</li> </ul>
Condicció de fi amb èxit	Validació transparent per l'usuari, el servidor no interromp l'acció.
Condicció de fi amb fracàs	El sistema NO autoritza l'ús de l'aplicació al client, mostrant un missatge d'error
Alternatives de procés	1- El sistema valida que l'ID de sessió de l'usuari sigui la més recent.
Excepcions	1.1- El sistema detecta que l'ID de sessió ha caducat, degut a que ja s'ha generat un ID més nou, i mostra un missatge d'error.
Altres comentaris	El control de sessió és similar al del client web, amb la diferència que el client web manté una sessió activa amb el servidor i en canvi Android realitza diferents peticions.

**Taula 20. Cas d'ús: control de sessió de l'usuari (Android)**

### 3.3.4.2. Agenda

Cas d'ús	b.1 Modificació d'esdeveniment
Descripció	Modificar l'estat o posar un comentari en un esdeveniment.
Actors	Usuari Android
Casos d'us relacionats	a.2 Control de sessió de l'usuari
Precondició	L'usuari s'ha autenticat correctament.
Condicció de fi amb èxit	Missatge amb el resultat de l'operació i s'actualitza l'agenda.
Condicció de fi amb fracàs	Missatge d'error al no trobar connexió amb el servidor
Alternatives de procés	1- L'usuari selecciona un esdeveniment del llistat. 2.a- L'usuari fa clic sobre el camp Estat. 2.a.1- L'usuari indica un Estat diferent a l'actual. 2.b- L'usuari fa clic sobre el quadre de text Comentaris. 2.b.1- L'usuari escriu un text i fa clic a Desar. 3- El sistema desa les dades al servidor 4- El sistema modifica l'indicador d'agenda modificada (marcador de data i hora) als usuaris assignats a l'esdeveniment. 5- El sistema retorna a l'aplicació Android les dades actualitzades de l'agenda i es mostra un missatge informatiu. 6- L'aplicació Android modifica l'indicador d'agenda modificada (marcador de data i hora) per a evitar recuperar dades innecessàries en properes consultes.

Excepcions	2.a.1/2.b.1- No hi ha connexió amb el servidor i es mostra un missatge d'error.
Altres comentaris	En cas de escollir l'Estat que ja estava assignat o posar un comentari en blanc, el sistema farà cas omís de l'acció.

**Taula 21. Cas d'ús: modificació d'esdeveniment (Android)**

Cas d'ús	b.2 Refrescar les dades de l'agenda
Descripció	L'usuari pretén obtenir l'agenda més recent.
Actors	Usuari Android
Casos d'us relacionats	a.2 Control de sessió de l'usuari
Precondició	L'usuari s'ha autenticat correctament.
Condicció de fi amb èxit	S'obre el navegador web per a mostrar la gestió de l'agenda.
Condicció de fi amb fracàs	Missatge d'error al no trobar connexió amb el servidor
Alternatives de procés	1- L'usuari fa clic sobre la icona Refrescar 2- El sistema compara el marcadore d'agenda modificada (data i hora) que consta a l'aplicació Android amb el marcadore que hi ha a base de dades. 3- El sistema retorna de nou a l'aplicació Android les dades de l'agenda en cas que hi hagi trobat diferències. 4- L'aplicació Android modifica l'indicador d'agenda modificada (marcadore de data i hora) per a evitar recuperar dades innecessàries en properes consultes.
Excepcions	1- No hi ha connexió amb el servidor i es mostra un missatge d'error.
Altres comentaris	

**Taula 22. Cas d'ús: refrescar les dades de l'agenda (Android)**



### 3.3.5. Prototipus de pantalles

#### 3.3.5.1. Benvinguda i autenticació

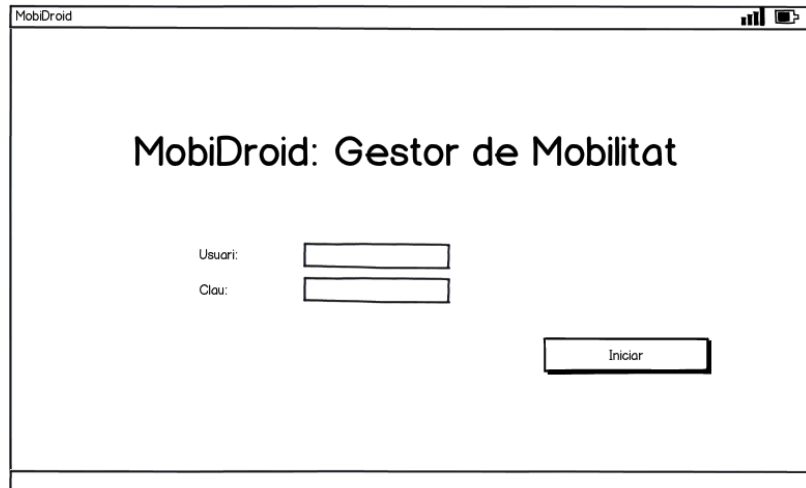


Fig. 14. Pantalles Android: benvinguda i autenticació

Aquesta pantalla haurà de contenir opcions de menú, que apareixerà al fer clic sobre el botó Menú del dispositiu si n'hi ha o fer clic sobre la icona d'Opcions, seguint les bones pràctiques d'Android. Inicialment, hi haurà l'opció de canviar el Host del servidor.

#### 3.3.5.2. Agenda (format tauleta)

La llista d'esdeveniments (esquerra), serà fixa. Al fer clic sobre la data ha d'aparèixer un selector de data. La part dreta contindrà el detall de l'agenda.

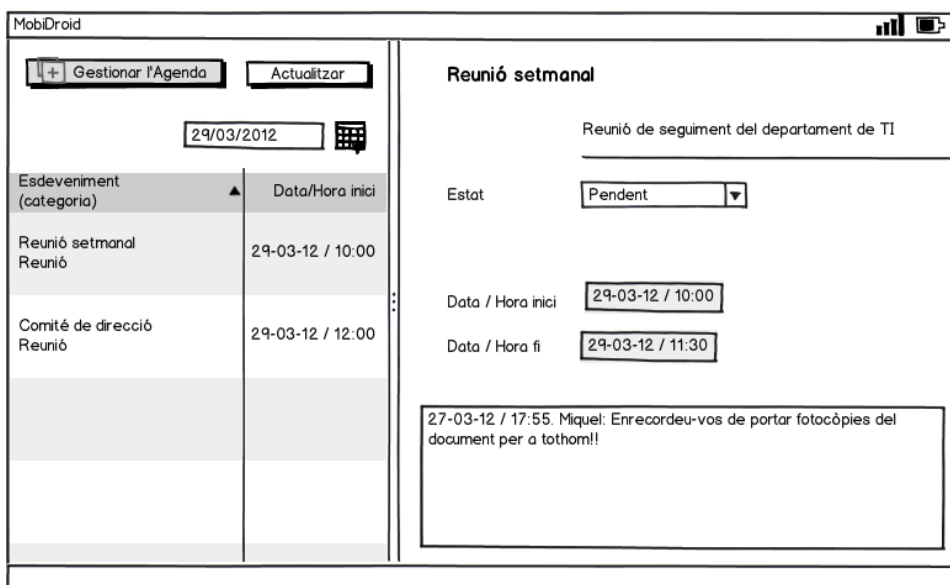


Fig. 15. Pantalles Android: agenda (format tauleta)

La navegació de la primera pantalla de detall a la segona serà fent ús de les propietats tàctils, podent lliscar amb el dit d'esquerra a dreta o a l'inrevés. Això hauria de permetre incorporar noves pantalles de detall en cas necessari.

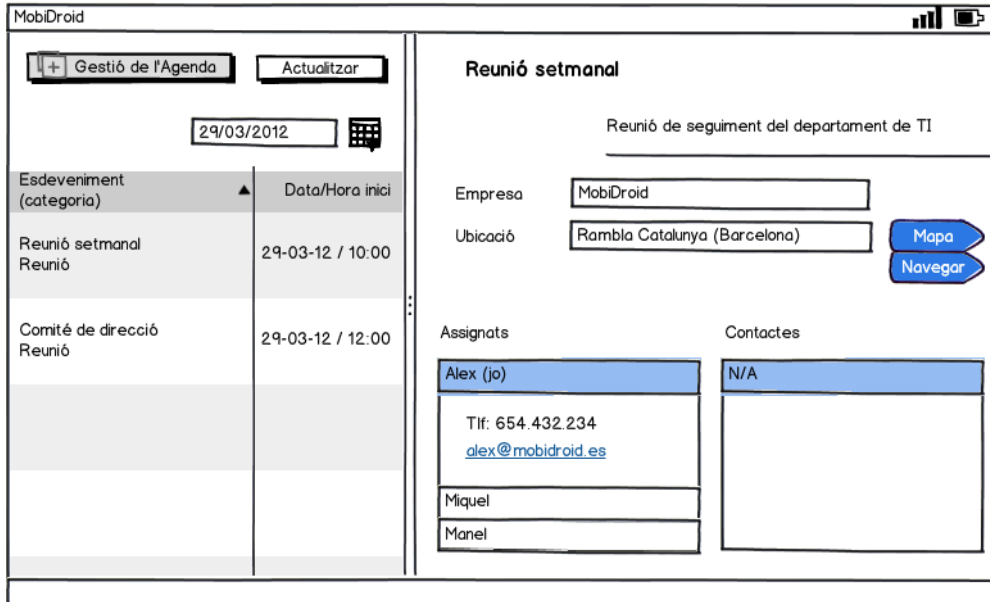


Fig. 16. Pantalles Android: agenda (2) (format tauleta)

Al fer clic sobre el telèfon es podrà trucar al número indicat, i al fer clic sobre el mail, preguntarà a l'usuari quin programa de missatgeria vol fer servir per a enviar el correu.

### 3.3.5.3. Agenda (format mòbil/vertical)

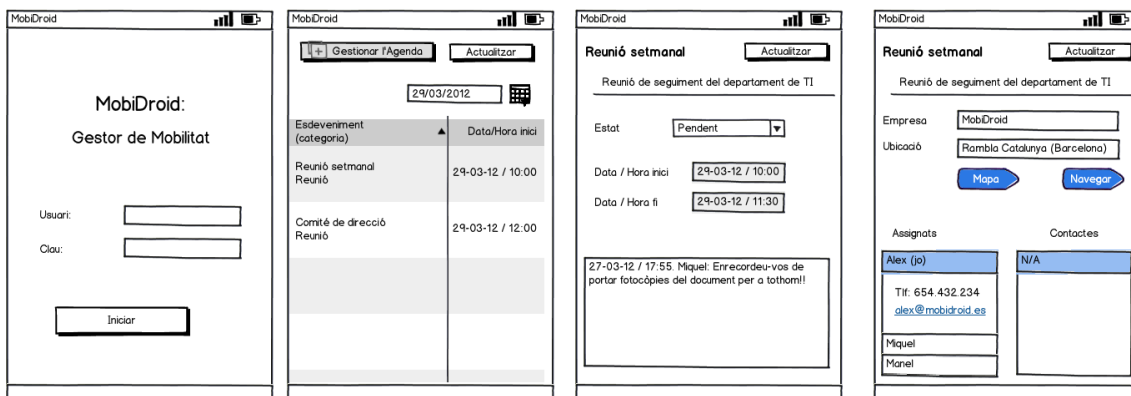


Fig. 17. Pantalles Android: agenda (format mòbil/vertical)

Es determinarà aquest format per la dimensió horitzontal de la resolució del dispositiu. En cas que la base (X) sigui d'una resolució igual o menor a 800dp, es passarà a treballar d'aquesta manera. La majoria de mòbils del mercat seguiran aquest format i també ho faran les tauletes en posició vertical, ja que per norma general la resolució horitzontal serà igual o menor a 800dp.

### 3.3.6. Propostes de millora

Del client Android també realitzaria moltes millores, nombro les següents:

- Procés en fons per a comprovar si hi ha dades noves. La periodicitat de la consulta hauria de ser configurable per l'usuari. Una vegada s'hagués detectat si hi ha dades noves, hauria de mostrar un missatge a l'usuari per a que executés voluntàriament el refresc de les dades, i a més, no hauria de realitzar noves consultes de dades noves a l'espera de l'acció de refresc per part de l'usuari.
- Mostrar missatges a la barra de notifiacions del mòbil. Aquest punt aniria lligat a l'anterior en quant a avisar que hi ha dades noves per actualitzar. També es podrien mostrar missatges d'alerta al detectar que falten X minuts per a que comenci un esdeveniment programat.
- S'hauria d'estudiar fins a quin punt s'ha de restringir el canvi d'estat. Per exemple, que l'usuari no pugui modificar un esdeveniment en estat Cancel·lat o Esborrat.
- Possibilitat de sincronitzar l'agenda amb Google Calendar.
- Possibilitat d'importar els contactes de MobiDroid als contactes del mòbil o tauleta.
- Possibilitat de que l'usuari informi de la seva ubicació, fent ús del GPS del mòbil o tauleta.
- Integrar Google Maps a l'aplicació, de mode que es pogués afegir una capa amb la informació de tots els esdeveniments alhora. Ajuntant el punt anterior, es podria mostrar al mapa la ubicació de tots els usuaris assignats a la tasca.
- Poder accedir a la informació de tots els usuaris i clients de MobiDroid, ja que actualment només es pot accedir a les dades dels que estan assignats a un esdeveniment.

## 4. Especificacions tècniques

### 4.1. Arquitectura del projecte

En aquest apartat comentaré breument els components utilitzats al projecte. La visió global es mostra en el següent diagrama:

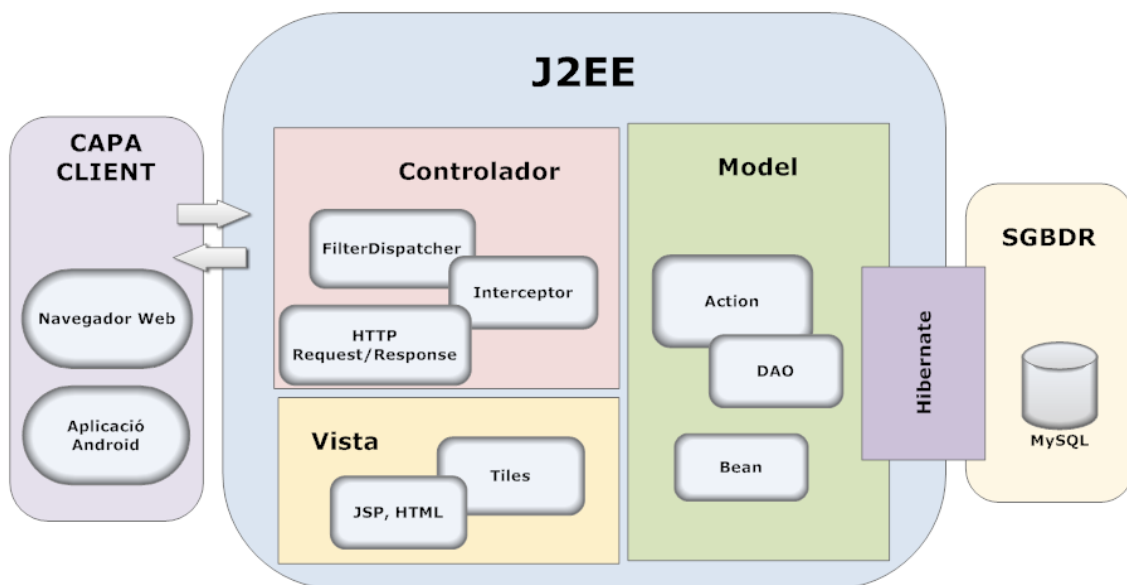


Fig. 18. Arquitectura del projecte

#### 4.1.1. J2EE

Java 2 Platform, Enterprise Edition és la plataforma de desenvolupament d'aplicacions distribuïdes i multicapes per a empreses. J2EE simplifica aquestes aplicacions fent servir components estàndards i modulars.

El llenguatge que utilitza és Java i, per tant, les aplicacions desenvolupades són multiplataforma. A més, proporciona APIs JDBC, per a les connexions amb bases de dades, i seguretat per a les comunicacions per Internet.

### **4.1.2. Apache Tomcat**

Tomcat és un senzill servidor Web de codi obert que suporta servlets i JSPs. Pot funcionar sobre qualsevol sistema que disposi de la màquina virtual Java.

Alternatives a Tomcat podrien ser JBoss o JOnAS, ambdós servidors d'aplicacions J2EE, però per les característiques del projecte finalment s'ha decidit utilitzar Tomcat.

### **4.1.3. MySQL Server**

MySQL és un sistema de gestió de base de dades relacional, amb molt bona reputació dins el sector. L'inconvenient respecte altres propostes és l'adquisició de llicències en cas d'un ús privatiu:

<http://www.mysql.com/about/legal/>

L'alternativa que contemplava es tractava de PostgreSQL, el SGBD utilitzat durant els estudis d'ETIG. La decisió de fer servir MySQL ha estat purament didàctica.

Per al modelat de la base de dades i preparació de jocs de proves, s'ha fet servir la utilitat MySQL Workbench.

### **4.1.4. Hibernate**

Per a connectar la base de dades amb el projecte J2EE s'ha fet servir el framework Hibernate, que proporciona un mapatge objecte/relacional (ORM), a més de facilitar la consulta i recuperació de dades.

### **4.1.5. Apache Struts2**

Struts2 és un framework per al desenvolupament Web que ens facilitarà l'implementació del patró MVC (Model-Vista-Controlador).

A destacar l'ús de tags propis i d'AJAX en els JSP, que simplifiquen molt la generació d'aquests fitxers al poder utilitzar directament classes POJO.

Struts2 proporciona les eines que ens permetran:

- gestionar les peticions del client.
- validar les dades, a client o al servidor.
- facilitar el control de totes les peticions, abans i després d'executar cada acció.
- executar l'acció sol·licitada, controlant el pas a seguir segons el resultat obtingut.
- capturar i proporcionar les dades a la vista.

#### **4.1.6. Apache Tiles**

Struts2 integra el plugin Apache Tiles, que es tracta d'un framework per a creació de plantilles de pàgines Web. Tiles permet descompondre el JSP i tractar-los independentment tot i que una vegada generat l'HTML es mostrin les diferents parts en la mateixa pantalla.

#### **4.1.7. SDK Android**

Per al desenvolupament del client Android es fa servir l'SDK distribuït per la marca. El llenguatge en el que es basa és Java i segueix el patró MVC, de manera similar a com es treballa per a la realització del client Web.

#### **4.1.8. Eclipse**

Eclipse Indigo ha estat l'entorn de desenvolupament (IDE) escollit, al haver-lo utilitzat a les diferents assignatures de programació d'ETIG.

#### **4.1.9. Versions**

- J2EE 6u4
- Apache Tomcat 7.0.26
- MySQL Server 5.5
- Hibernate 3
- Struts2 2.3.1
- Apache Tiles 2.0.6
- SDK Android 4.0.3
- Eclipse Indigo 3.7.0

## 4.2. Disseny de la persistència

En el model de base de dades diferenciarem les taules de dades mestres i les taules amb dades transaccionals. Les de dades mestres, seran les següents:

- Usuari [Usuari]: usuaris del sistema, identificats pel seu NIF
- Client [Client]: empreses que poden ser objectiu d'algun esdeveniment. El CIF o NIF no serà el camp clau, per a permetre entrades quan no es coneix aquesta dada.
- Contacte [Contacte]: usuaris de les empreses que poden ser assignats a un esdeveniment. De la mateixa manera que el Client, el contacte no s'identificarà pel NIF. En aquest cas, tampoc l'enregistrarem ja que no el necessitarem.

Les taules transaccionals, són:

- Agenda [Agenda]: taula on s'enregistraran les dades principals dels esdeveniments.
- Usuari participant [UsuAgenda]: assignacions dels usuaris als esdeveniments.
- Contacte participant [ConAgenda]: assignacions dels contactes als esdeveniments.
- Comentari [Comentari]: comentaris dels usuaris en els esdeveniments.

En el diagrama d'Entitat-Relació s'estableixen les relacions entre les taules:

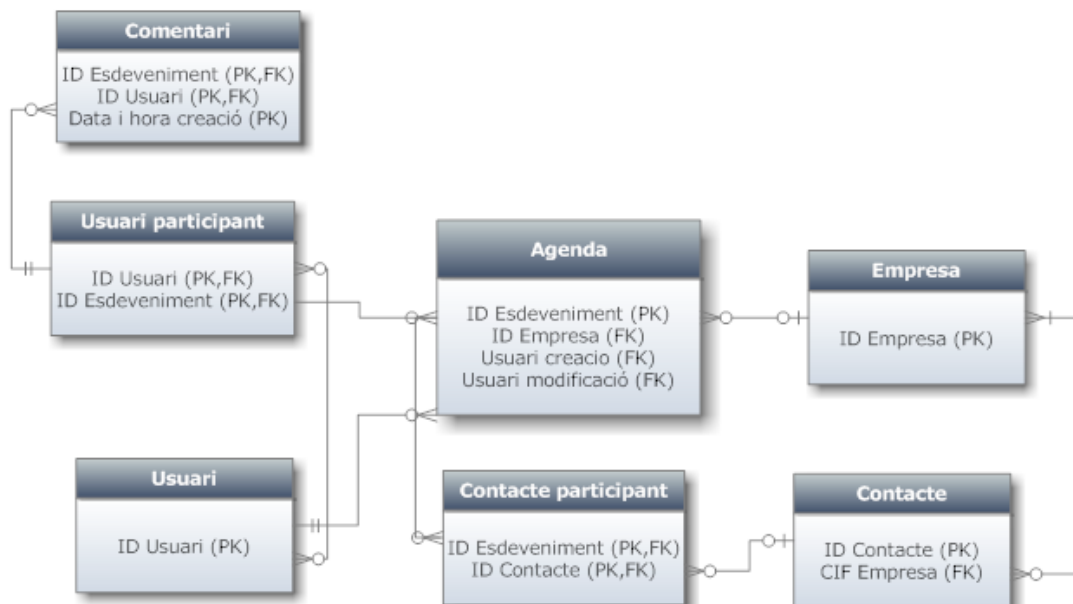


Fig. 19. Diagrama d'entitat-relació

Només s'han detallat els camps clau i les claus foranies per a simplificar el diagrama.

#### 4.2.1. Atributs

**Usuari**(NIE, Nom, Cognoms, Perfil, Departament, Telefon, Mail, Contrasenya, Actiu, IdSessio, UsuCre, DataHoraCre, UsuMod, DataHoraMod, DataHoraAgMod)

On { UsuCre } és clau forana a Usuari(NIF)

On { UsuMod } és clau forana a Usuari(NIF)

**Client**(Id, Nom, CIF, UsuCre, Pais, Poblacio, Direccio, DataHoraCre, UsuMod, DataHoraMod)

On { UsuCre } és clau forana a Usuari(NIF)

On { UsuMod } és clau forana a Usuari(NIF)

**Contacte**(Id, Client, Nom, Cognoms, Telefon, Mail, UsuCre, DataHoraCre, UsuMod, DataHoraMod)

On { Client } és clau forana a Client (Id)

On { UsuCre } és clau forana a Usuari(NIF)

On { UsuMod } és clau forana a Usuari(NIF)

**Agenda**(EsdevenimentId, Categoria, Titol, Estat, DataHoraIni, DataHoraFi, Client, Pais, Poblacio, Direccio, Descripcio, UsuCre, DataHoraCre, UsuMod, DataHoraMod)

On { Client } és clau forana a Client (Id)

On { UsuCre } és clau forana a Usuari(NIF)

On { UsuMod } és clau forana a Usuari(NIF)

**UsuAgenda**(Usuari, Esdeveniment)

On { Usuari } és clau forana a Usuari(NIF)

On { Esdeveniment } és clau forana a Agenda(EsdevenimentId)

**ConAgenda**(Esdeveniment, Contacte )

On { Esdeveniment } és clau forana a Agenda(EsdevenimentId)

On { Contacte } és clau forana a Contacte(Id)

**ComAgenda**(Usuari, Esdeveniment, DataHoraCre, Comentari)

On { Esdeveniment } és clau forana a Agenda(EsdevenimentId)

On { Contacte } és clau forana a Contacte(Id)



### 4.3. Projecte Web

#### 4.3.1. Recursos

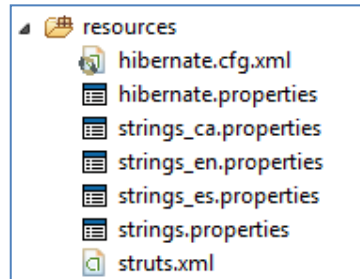


Fig. 20. Recursos del projecte Web

El fitxer **hibernate.cfg.xml** contindrà les configuracions per a fer possible el mapatge i connexió de les classes d'entitat amb MySQL. Els fitxers **strings\_<idioma>** contindrà els textos que farà servir el client web. Per últim, el fitxer **struts.xml** contindrà la configuració d'Struts2.

#### 4.3.2. Estructura de paquets

Es prepararà la següent estructura de paquets, que facilitaràn l'ús del patró MVC:

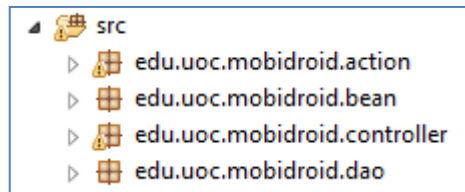


Fig. 21. Estructura de paquets del projecte Web

Al paquet **action** estaran les classes destinades a realitzar la lògica de negoci segons el model d'Struts2, i seguint aquest model farem servir el paquet **controller** amb les classes de control i filtrat. D'aquestes classes i d'Struts2 en parlarem més endavant.

El paquet **bean** contindrà les classes d'entitat, i per a obtenir la informació d'aquestes classes es faran servir les classes DAO del paquet **dao**. Aquests dos paquets quedaran detallats als diagrames de classes corresponents.

### 4.3.3. Diagrames de classes

#### 4.3.3.1. Diagrama de classes d'entitat

En el diagrama s'especificaran els atributs de les classes entitat:

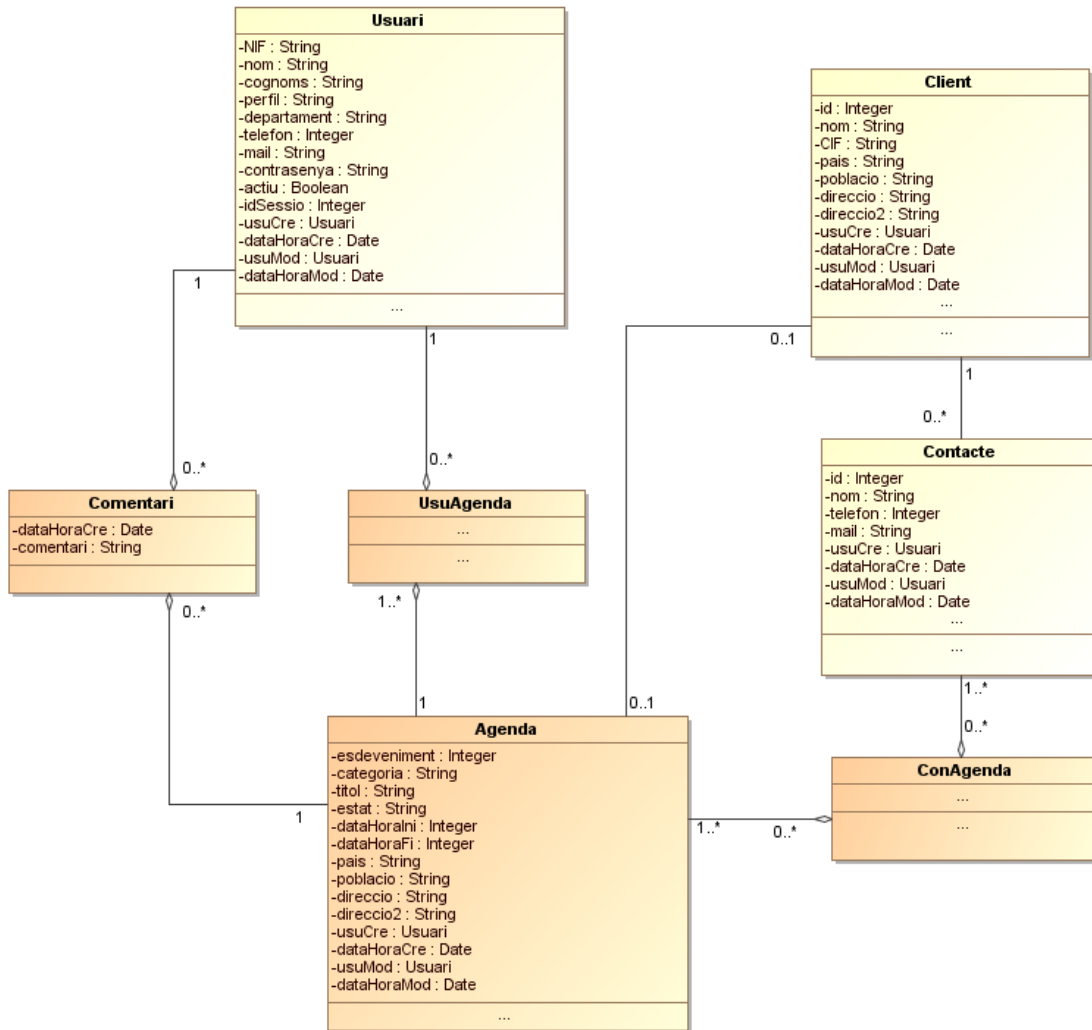


Fig. 22. Diagrama de classes d'entitat

### 4.3.3.2. Diagrama de classes DAO

Passem a detallar els mètodes proporcionats per les classes DAO del projecte:

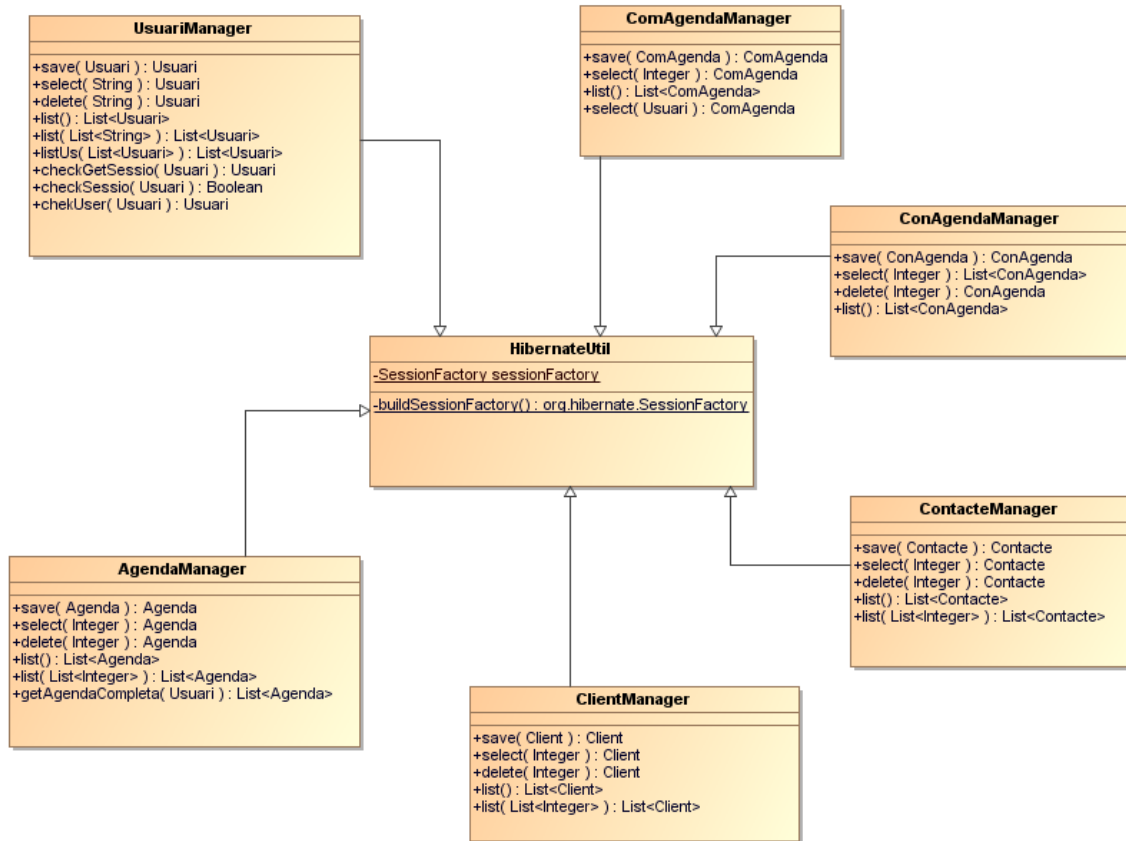


Fig. 23. Diagrama de classes DAO

### 4.3.4. Implementació d'Struts2

L'arquitectura Struts2 a MobiDroid es mostra al diagrama d'aquesta pàgina, adaptat als requeriments del projecte. L'original de Struts2 es pot consultar a la seva pàgina web:

<http://struts.apache.org/2.x/docs/big-picture.data/Struts2-Architecture.png>

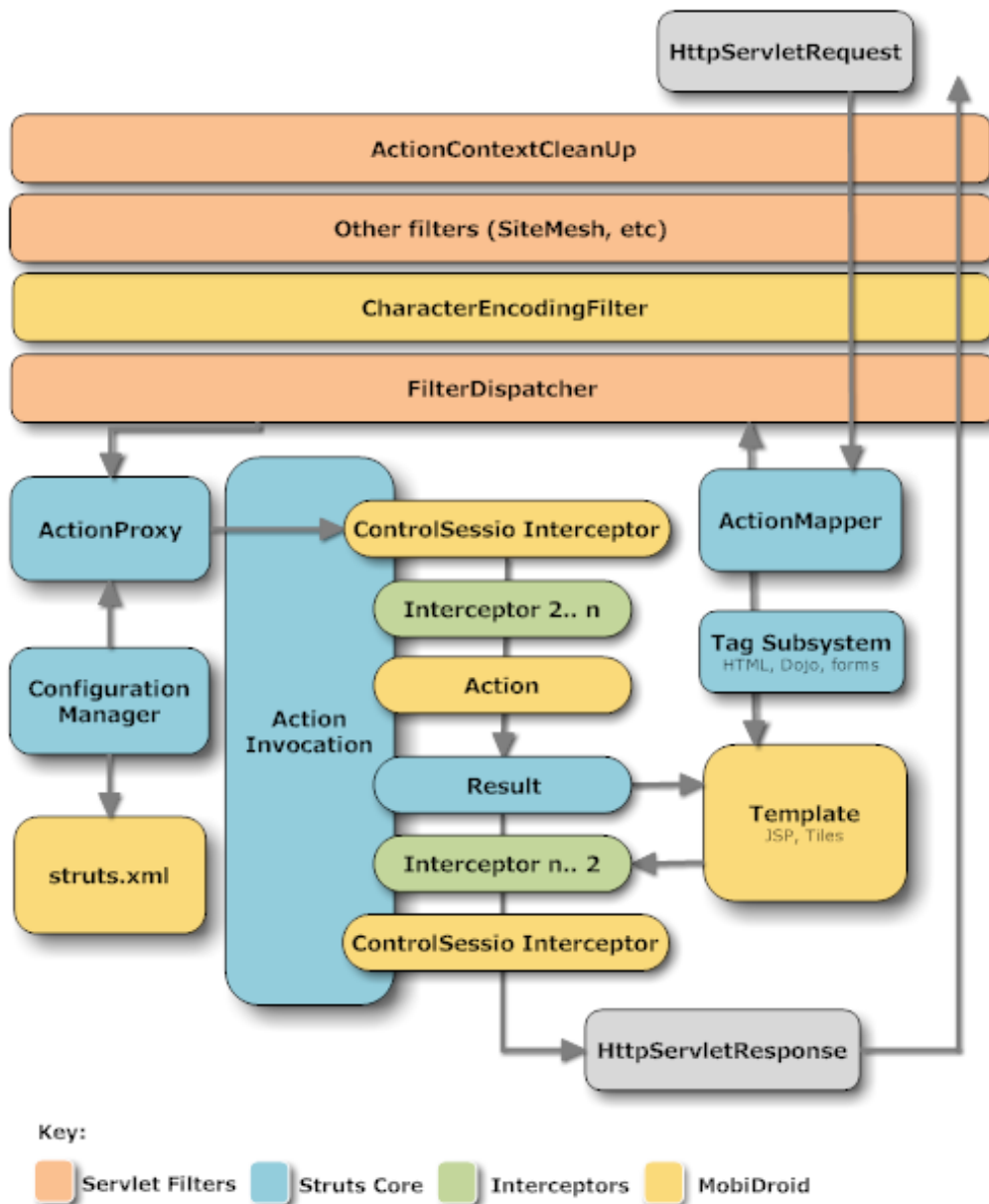


Fig. 24. Implementació d'Struts2

En el diagrama, es reflexa com la petició del client passa per una sèrie de filtres estàndards i pel filtre **CharacterEncodingFilter**, declarats al **web.xml**. Aquest filtre s'ha hagut d'integrar degut a problemes amb la codificació dels caràcters especials a UTF8, que provocava la pèrdua dels

mateixos al desar les dades a la base de dades. El codi d'aquesta classe s'ha aconseguit de la pàgina web:

<http://mayankgsingh.blogspot.com.es/2009/06/special-characters-in-web-application.html>

A continuació, **FilterDispatcher** consulta l'**ActionMapper** per determinar les accions a invocar i delega el control al **ActionProxy**, que consulta l'**struts.xml** i crea l'**ActionInvocation**.

A partir d'aquí s'executaran els **Interceptors**, començant pel **controlSessioInterceptor**, classe creada per controlar la sessió de l'usuari, i seguint pels definits per defecte al Struts2. Després dels *Interceptors* passa a executar-se l'**Action** desitjat, obtenint un **Result** que juntament amb el mapatge realitzat a **struts.xml** permetrà dictaminar si s'ha de redirigir cap algun dels **JSP** o cap a una altra **Action**. Cal dir que l'Interceptor pot evitar l'execució de l'**Action** al retornar un **Result** diferent de **SUCCESS** (resultat esperat).

Per últim, es tornen a executar els *Interceptors* en ordre invers i retornant una resposta a través dels filtres declarats al **web.xml**.

En el projecte conviuran les *Action* destinades a proporcionar informació als JSP i l'*Action* UsuariJSONAction, que serà la que s'utilitzarà per a les connexions amb el client Android o que retornarà el resultat en format **JSON**.

#### 4.3.4.1. Patró MVC

S'ha seguit el patró MVC definit per Struts2:

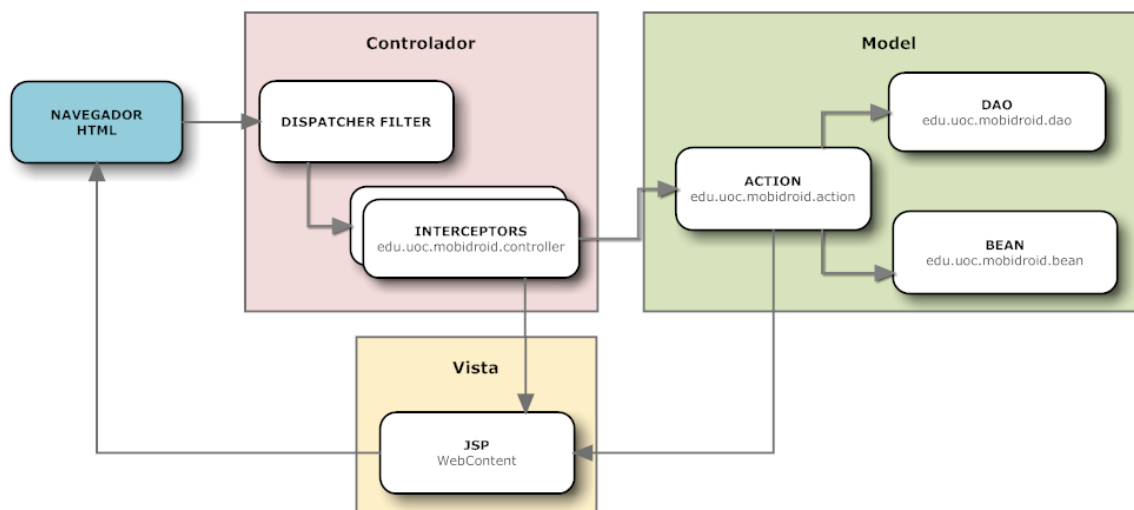


Fig. 25. Patró MVC

#### 4.3.4.2. JSP i Tiles

Els JSP (Java Server Pages) permeten generar dinàmicament l'HTML que serà executat des del navegador Web. Els fitxers quedaran repartits segons la definició del **tiles.xml**, ubicat a WebContent/WEB-INF.

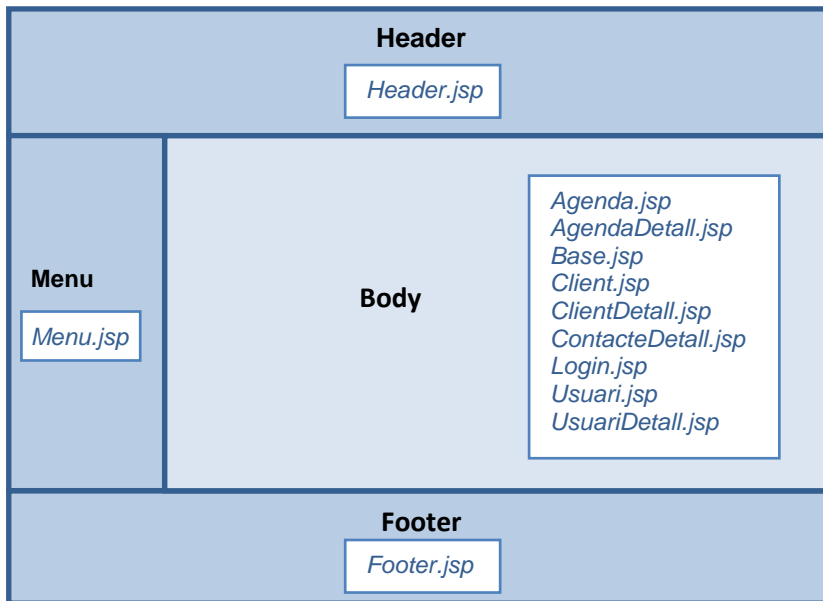


Fig. 26. Esquema JSP i Tiles

#### 4.3.4.3. Estils CSS

S'han definit una fulla d'estil per a la pantalla `Login.jsp` i un altre per a la resta de la web.

#### 4.3.4.4. Internacionalització (i18n)

S'utilitzaran els fitxers següents per a la gestió dels textos de la web:

- `strings.properties`. Textos per defecte, en català
- `strings_ca.properties`. Textos en català
- `strings_es.properties`. Textos en castellà
- `strings_en.properties`. Textos en anglès

A la pantalla d'autenticació es podrà seleccionar l'idioma, seguint els estàndards de Struts2.

#### 4.3.4.5. Control de sessió del client Web

La classe `ControlSessioInterceptor` s'encarregarà de controlar la sessió al client Web, i serà cridada abans de cada *Action* amb les excepcions següents:

- Accions per a l'autenticació
- Accions per a la comunicació amb Android
- Action per a la gestió de l'idioma

Què validarà?

1. L'existència de l'objecte booleà *Logged-in* i amb valor *TRUE*. Aquest objecte es genera en el moment d'autenticar-se i es destrueix en el moment de tancar la sessió (botó Desconnectar).
2. L'existència de l'ID de sessió i la caducitat d'aquest ID. Aquest ID es tracta de la concatenació de data i hora i es genera en el moment d'autenticar-se, desant-se també a la taula *Usuari* de la base de dades *MobiDroid*. Per tant, es realitzarà la consulta per a comprovar si aquest ID és el correcte.

En cas d'error, la classe retornarà el text *login*, i gràcies al mapatge d'aquest resultat a l'*struts.xml*, l'usuari serà redirigit a la pantalla d'autenticació, mostrant el missatge d'error pertinent.

#### 4.3.4.6. Validacions a client i a servidor

Struts2 ens permet diferenciar les validacions de format de les validacions de negoci. A més, tenim l'opció de validar format directament a client, amb la millora de rendiment que comporta. En el projecte farem servir aquesta funcionalitat en els formularis que permeten desar les dades al servidor.

Per a les validacions a client farem servir els fitxers XML:

- ✓ Camps obligatoris
- ✓ Format dels camps numèrics
- ✓ Longitud mínima dels NIF/CIF i de les contrasenyes d'usuari
- ✓ Format dels mails

Al servidor validarem:

- ✓ L'existència d'usuaris assignats al desar d'un esdeveniment.
- ✓ Format del NIF espanyol.

## 4.4. Projecte Android

### 4.4.1. Recursos

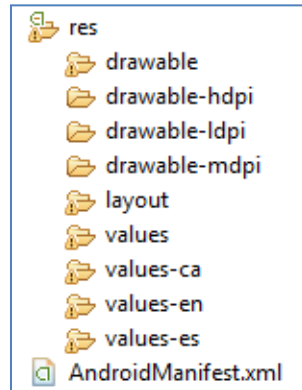


Fig. 27. Recursos del projecte Android

Dels recursos del projecte Android destaquem:

- Carpetes *drawable*: icones de l'aplicació
- Carpeta *layout*: pantalles en format XML
- Carpetes *values*: fitxers XML amb els textos que es faran servir, tan per a etiquetes com per definir els colors.
- *AndroidManifest.xml*: fitxer amb les declaracions de l'aplicació, on s'estableix la versió mínima i la recomanada del sistema operatiu Android necessari per a fer funcionar l'aplicació, així com els recursos que necessitarà del dispositiu i els permisos que haurà de demanar a l'usuari. Altres propietats que es poden definir són l'icona principal, el títol de l'aplicació i les classes *Activity* que s'executaran.

### 4.4.2. Estructura de paquets

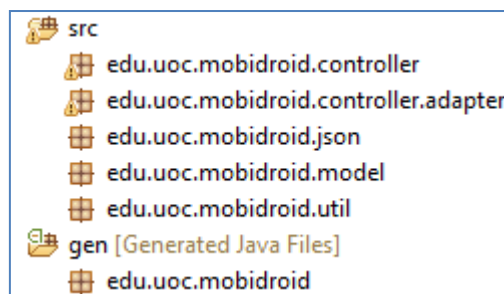


Fig. 28. Diagrama de paquets del projecte Android

Els paquets contindran les classes següents:

- *edu.uoc.mobidroid*: classe *R.java* autogenerada per Android, que conté referències a tots els recursos de l'aplicació.



- `edu.uoc.mobidroid.controller`. Conté les classes *Activity*, que són les que informen els elements que es mostraran als *layouts*. Disposen d'una sèrie d'events per a poder definir la lògica de l'aplicació en funció de l'acció realitzada per l'usuari.
- `edu.uoc.mobidroid.controller.adapter`. Classes que ens serviran de suport per a les *Activity* a l'hora d'alimentar components de l'aplicació, com el selector d'agendes o el control de la paginació de pantalles.
- `edu.uoc.mobidroid.json`. Classes per a la comunicació amb el servidor. Les comentarem en el punt següent.
- `edu.uoc.mobidroid.model`. Classes bean, idèntiques a les creades en el projecte Web per a facilitar la comunicació amb el servidor mitjançant JSON.
- `edu.uoc.mobidroid.util`. Classes amb mètodes reutilitzables. Destaquem la classe `Preferencies.java`, per a la gestió de les preferències de l'usuari, que es desaran físicament en un fitxer de l'aplicació.

#### 4.4.3. Serveis Web. JSON

Per a la connexió de l'aplicació Android amb el servidor, m'he decantat per utilitzar el format JSON (JavaScript Object Notation). Aquest format estableix unes especificacions per a emmagatzemar les dades en format de text pla. El receptor de la petició ha de saber com interpretar el contingut rebut, però hi ha mètodes eficients i senzilles d'automatitzar aquest procés. La solució presa és l'ús de la llibreria **GSON** de Google, que ens permet serialitzar i deserialitzar amb molta facilitat a partir dels mètodes següents:

- `toJson(<instància de la classe a serialitzar>, <Nom de la classe.class>)`
- `fromJson(<text del json a deserialitzar>, <Nom de la classe.class>)`

Aprofitarem aquest recurs per a cridar els diferents serveis web enviant i rebent la instància d'Usuari amb tota la informació requerida.

Per a generar el contingut del text en format JSON, cridarem al mètode `toJson` passant per paràmetre la instància de la classe `Usuari`, que sempre contindrà el NIF i la clau (si el servei cridat és el d'autenticació) o l'ID de sessió (si ja ens hem autenticat).

Al finalitzar el servei, rebrem de nou un text en format JSON i deserialitzarem realitzant un `cast` també en un objecte de la classe `Usuari`.

Com recuperarem la resta de dades i com desarem? La instància d'Usuari també tindrà com atribut una llista de la classe `Agenda`, que la farem servir per a recuperar l'agenda completa de

l'usuari una vegada s'ha autenticat i també per a fer viatjar les modificacions dels camps Estat i Comentaris.

Els serveis JSON implementats (definites al projecte Web) són aquests:

- `jsonLogin`: servei per a l'autenticació de l'usuari i obtenir les dades de l'agenda.
- `jsonComentari`: servei per a desar els comentaris de l'usuari al servidor.
- `jsonRefresh`: servei per a recuperar les dades de l'agenda i refrescar la pantalla, si és que n'ha trobat de noves.
- `jsonEstat`: servei per a desar el canvi d'estat d'un esdeveniment al servidor.

#### 4.4.4. Composició dels Layouts

Els layouts d'Android ofereixen la possibilitat de dividir-los en petites parts, que permeten la seva reutilització. S'ha aprofitat aquesta propietat, de tal manera que ha estat força senzill implementar dues maneres de navegar (comentat al punt 3.3.5 *Prototipus de pantalles*), depenent de la resolució de la pantalla del dispositiu utilitzat.

#### 4.4.5. Internacionalització (i18n)

Android té una gestió molt senzilla dels idiomes, ja que es basa en el nom de la carpeta per saber l'idioma dels textos del fitxer `strings.xml`. Al projecte tindrem:

- `values/strings.xml`. Textos per defecte, en català
- `values_ca/strings.xml`. Textos en català
- `values_es/strings.xml`. Textos en castellà
- `values_en/strings.xml`. Textos en anglès

L'aplicació pren per defecte l'idioma del Sistema Operatiu. Al tractar-se de dispositius mòbils entenem l'usuari té configurat el seu idioma i, per tant, l'opció de canviar-lo no s'ha implementat al no considerar-lo prioritari.

#### 4.5. Valoració econòmica

La valoració econòmica s'ha realitzat sota el supòsit de ser una empresa consultora venedora de programari. El preu per recurs és aproximat a la realitat, i els dies dedicats són els de la vida d'aquest projecte però amb una dedicació de 8 hores. No s'ha tingut en compte l'adquisició de maquinari ni altres recursos per part de l'empresa.

Recursos	€/Dia	€/hora
Gerent	448	56
Consultor	248	31
Analista programador	200	25
Programador	160	20

Dies
Import (Dies x Preu/Dies)

Tasca	Gerent		Consultor		Analista		Programador		Total
Presa de requeriments	5	2240	9	2232	2	400			4872
<b>Total FASE 1</b>	5	2240	9	2232	2	400			4872
Proves dels frameworks					2	400	7	1120	1520
Anàlisis funcional			7	1736	2	400			2136
Dissenys tècnics					4	800			800
Preparació dels entorns					5	1000	3	480	1480
<b>TOTAL FASE 2</b>	0	0	7	1736	12	2400	17	2720	5936
Modelat de la base de dades					1	200	1	160	360
Desenvolupament					8	1600	12	1920	3520
Fase de proves					2,5	500	3,5	560	1060
Correccions					3	600	9	1440	2040
Millores en la interfície gràfica						0			0
<b>Total FASE 3</b>	0	0	0	0	14,5	2900	25,5	4080	6980
Redacció i revisió de la memòria		0	2,5	620	1,5	300	6	960	1880
Creació i revisió de la presentació			1,5	372	0,5	100	2	320	792
<b>Total FASE 4</b>	0	0	4	992	2	400	8	1280	2672
<b>Total projecte</b>								<b>20.460 €</b>	

Taula 23. Valoració econòmica

## 4.6. Conclusions

Una vegada finalitzat el projecte, és l'hora de treure conclusions. Revisant els objectius principals que em vaig fixar, penso que els puc marcar pràcticament tots com a complerts. Ha estat un projecte molt dur, com ja havia previst des del principi. A part dels objectius, voldria comentar els aspectes més positius:

- Haver après J2EE m'ha reconfortat molt més del que hem pensava abans d'escollir aquesta temàtica de TFC. Fins i tot opino que hauria de ser una assignatura més dins la carrera d'informàtica, per tots els conceptes nous i interessants que aporta i que de no haver realitzat el projecte m'hagués quedat sense aprendre.
- Surto recolzat moralment, per haver-me demostrat a mi mateix que puc tirar cap endavant amb noves tecnologies d'una manera autodidacta i en tan poc temps.
- Executar l'aplicació final i veure que funciona pràcticament com l'havia imaginat en un primer moment (exceptuant canvis no significatius), em fa pensar que he seguit les pautes correctes.
- Programar amb Android, tot i que no era objectiu prioritari del TFC, ha estat molt divertit, amb resultats ràpids i amb ganes de continuar aprenent aquesta tecnologia. Ajuntant J2EE i Android, poden sorgir projectes molt ambiciosos.

Sent crítics, hi ha coses que podria haver millorat. Aquestes són:

- Haver deixat coixins de temps més amplis a la planificació, per poder cobrir alteracions en la data d'inici d'algunes tasques.
- Posar-me uns objectius funcionals menys optimistes, sobretot perquè vaig dedicar més hores de les previstes al client Android en detriment d'haver après altres tècniques i llenguatges de programació com podien ser Ajax, JQuery o HTML5.

## 5. Glossari

<i>Action</i>	Tipus de classe d'Struts2 que realitza l'acció sol·licitada per l'usuari o sistema, i és on s'implementa la lògica de negoci de l'aplicatiu.
<i>Activity</i>	Tipus de classe d'Android que, d'una manera similar a les Action, s'encarrega del model de negoci de l'aplicació, proporcionant també les dades que es mostraran per la pantalla
<i>AJAX</i>	Són les sigles de Asynchronous Javascript And Xml, (javascript asíncron i XML), un conjunt de tecnologies que permeten actualitzar continguts web sense haver de tornar a carregar la pàgina.
<i>Backup</i>	En català, còpia de seguretat. Terme que s'acostuma a utilitzar a base de dades, proporcionant l'opció de restaurar una còpia de les dades anteriors en cas necessari.
<i>Base de dades Relacional</i>	És una base de dades que compleix amb el model relacional, el qual és el model més utilitzat en l'actualitat per a modelar problemes reals i administrar dades dinàmicament
<i>Bean</i>	Terme que es fa servir per a catalogar les classes que representen una realitat del model de l'aplicació, com per exemple: persona, cotxe..
<i>Cas D'ús</i>	És una tècnica per a la captura de requisits potencials d'un nou sistema o una actualització de programari.
<i>Codi obert</i>	De vegades referit com a "codi font obert" (open source en anglès) és un enfocament per al disseny, el desenvolupament i la distribució que ofereix un accés pràctic al codi font del producte (béns i coneixements).
<i>Col·laboratiu</i>	Adjectiu utilitzat per a catalogar les eines empresarials que permeten la interacció entre els usuaris
<i>CSS</i>	Cascading Style Sheets (CSS, en català Fulls d'Estil en Cascada) és un llenguatge de fulls d'estil utilitzat per descriure la semàntica de presentació (l'aspecte i format) d'un document escrit en un llenguatge de marques.
<i>DAO</i>	Data Acces Object. Tipus de classe que implementa els mecanismes per accedir a les bases de dades
<i>Drawable</i>	A Android, carpeta del projecte que conté les imatges
<i>Escalable</i>	Propietat que indica que un mètode, classe o programa pot ser ampliat amb relativa facilitat
<i>Framework</i>	És un terme adoptat de l'anglès i equival a entorn de treball o, també, marc de treball.
<i>GSON</i>	Llibreria Java que facilita la serialització i deserialització en format JSON.
<i>Host</i>	En català, servidor. Fa referència a la IP que identifica a la màquina utilitzada com a servidor.
<i>HTML</i>	(acrònim d'Hyper Text Markup Language, en català, "llenguatge de marcat d'hipertext"), és un llenguatge de marcat que deriva de l'sgml dissenyat per estructurar textos i relacionar-los en forma d'hipertext.
<i>i18n</i>	Estàndard per a la internacionalització i localització
<i>IDE</i>	És l'acrònim anglès per a denominar un entorn integrat de desenvolupament.
<i>Interceptor</i>	Tipus de classe d'Struts2 que s'executa abans i després d'una Action, amb la funció de controlar o validar
<i>J2ee</i>	Java Platform, Enterprise Edition o Java EE (va ser conegut com Java 2 Platform Enterprise Edition o J2EE fins la versió 1.4), és una plataforma de programació (una de les Plataformes Java) per desenvolupar i executar programari escrit amb el llenguatge Java amb una arquitectura distribuïda amb nivells, basada en components de programari, tot plegat executant-se en un servidor d'aplicacions.

<i>Javadoc</i>	Utilitat de Java que estandarditza la documentació del codi de l'aplicació, podent exportar-la a format web.
<i>jQuery</i>	jQuery és una biblioteca o framework de Javascript, creada inicialment per John Resig, que permet simplificar la manera d'interactuar amb els documents HTML, manipular l'arbre DOM, gestionar esdeveniments, desenvolupar animacions i afegir interacció amb la tecnologia AJAX en pàgines web.
<i>JSON</i>	(acrònim de javascript Object Notation) és un estàndard obert basat en text dissenyat per a intercanvi de dades llegible per humans. S'utilitza principalment per intercanviar dades entre un servidor i una aplicació web, sent una alternativa a l'xml
<i>JSP</i>	Javascript Pages és una tecnologia que permet als desenvolupadors de pàgines web, generar respostes dinàmicament a peticions HTTP.
<i>Layout</i>	A Android, arxiu que conté la disposició dels elements de la pantalla
<i>Look and Feel</i>	En català, vista i sensació, es refereix al disseny visual del programari
<i>Màquina Virtual</i>	És un programari que emula un ordinador i pot executar programes com si fos un ordinador real.
<i>Modular</i>	Adjectiu utilitzat per a indicar que un mètode, classe o programari està creat de forma descentralitzada, en components, de manera que la modificació d'un d'ells no hauria d'afectar a la resta.
<i>Mvc</i>	Model-Vista-Controlador, és un patró de disseny per al desenvolupament de programari que separa el model de dades, la interfície usuari i la lògica de control.
<i>Perfil</i>	Conjunt de característiques que defineixen a un usuari, permetent agrupar-los i aplicar diferències, sobretot pel que fa a autoritzacions
<i>Portable</i>	Adjectiu utilitzat per a indicar que un programari pot ser executat sota diferents sistemes operatius o dispositius
<i>Precondició</i>	Condicció necessària per a que les accions d'un cas d'ús puguin ser realitzades
<i>Programari lliure</i>	El programari lliure (en anglès free software) és el programari que pot ser usat, estudiat i modificat sense restriccions, i que pot ser copiat i redistribuït bé en una versió modificada o sense modificar sense cap restricció, o bé amb unes restriccions mínimes per garantir que els futurs destinataris també tindran aquests drets.
<i>Resolució</i>	Característica per a identificar la quantitat de píxels (unitat mínima que forma una imatge gràfica) que es mostraran per pantalla. A més quantitat de píxels, més resolució i millor qualitat de visionat
<i>Serialitzar</i>	A Java, convertir un format interpretable en un format estandarditzat però que no es pot interpretar. L'acció contrària s'anomena deserialitzar.
<i>Servidor d'aplicacions</i>	Un servidor d'aplicacions és un motor que ofereix serveis d'aplicacions a clients o dispositius. A més, un servidor d'aplicacions se'n cuida de la majoria, sinó tota, la lògica de negoci i accés a les dades de l'aplicació.
<i>Servidor Web</i>	Ordinador que accepta requeriments HTTP de clients, normalment coneguts com a navegadors web, i de servir-los pàgines web, que normalment són documents HTML.
<i>Servlet</i>	Les miniaplicacions de servidor (anglès servlets) són objectes Java executats per un servidor d'aplicacions i que responen a invocacions HTTP, servint pàgines dinàmiques.
<i>Tag</i>	L'etiqueta (en anglès tag) és una paraula o conjunt de paraules que defineixen d'una forma clara i senzilla ja sigui l'article escrit, la imatge publicada, el document, dins d'un web

Per a la confecció del glossari m'he ajudat de la web: <http://ca.wikipedia.org>

## 6. Bibliografia

### WEB

eFunctions: <http://efunctions.wordpress.com/>

Tutoriales: [www.adictosaltrabajo.com/](http://www.adictosaltrabajo.com/)

Struts2: <http://mundogeek.net/>

Struts 2 Tiles Plugin Tutorial with Example in Eclipse: <http://viralpatel.net/blogs/>

Dojo toolkit: <http://es.wikipedia.org/>

Sobre Struts 2: <http://struts.apache.org/>

Hibernate: <http://www.vaannila.com/>

Instalar Hibernate: <http://thelordofthebits.com/>

Google Developers: <https://developers.google.com/>

JSON Support: <http://jersey.java.net/>

Community Projects: <http://www.jboss.org/projects/>

Manual web: <http://www.manualweb.net/>

Login interceptor: <http://www.vitarara.org/>

Special characters in web application: <http://mayankgsingh.blogspot.com.es>

Model View Controller Pattern: <http://best-practice-software-engineering.ifs.tuwien.ac.at/>

### ANDROID

Android Developers: <http://developer.android.com/index.html>

Guardar leer preferencias de configuración aplicación Android SharedPreferences:  
<http://www.ajpdsoft.com/>

Dojo tip stricks: <http://dojotipsntricks.com/>

WikiDroid: <http://www.wikidroid.es/>

ListView en Android: <http://www.nosolounix.com/>

Vogella: <http://www.vogella.com/>

How-to: Android as a RESTful Client: <http://senior.ceng.metu.edu.tr/2012/>

Calling Web Services in Android using HttpClient : <http://lukencode.com/>

Android accessing RESTFull Web Services using JSON: <http://www.josecgomez.com/>

Taller: Consumiendo servicios Web desde Android: <http://www.andronautas.com/>

Android: Parsing JSON with Generics: <http://vladexologija.blogspot.com.es/>