

Títol del Treball Final

Josep Benet Borrajo Fernàndez
Grau d'Enginyeria Informàtica
Àrea de treball final

Nom Consultor/a
Albret Grau Perisé

12/06/2020



Aquesta obra està subjecta a una llicència de [Reconeixement-SenseObraDerivada 3.0 Espanya de Creative Commons](https://creativecommons.org/licenses/by-nc/3.0/)

FITXA DEL TREBALL FINAL

Títol del treball:	<i>Intranet Ubilibet 2.0</i>
Nom de l'autor:	<i>Josep Benet Borrajo Fernàndez</i>
Nom del consultor/a:	<i>Albert Grau Perisé</i>
Nom del PRA:	<i>Santi Caballé Llobet</i>
Data de lliurament (mm/aaaa):	<i>06/2020</i>
Titulació o programa:	<i>Grau d'Enginyeria Informàtica</i>
Àrea del Treball Final:	<i>TGF – Java EE</i>
Idioma del treball:	<i>Català</i>
Paraules clau	<i>Màxim 3 paraules clau, validades pel director del treball (donades als estudiants o en base a llistats, thesaurus, etc.)</i>

Resum del Treball (màxim 250 paraules): *Amb la finalitat, context d'aplicació, metodologia, resultats i conclusions del treball*

Projecte per una nova versió d'Intranet per a l'empresa Ubilibet S.L. amb l'objectiu de dotar d'un sistema de permisos per a usuaris. Realitzada amb Java EE, amb base de dades Postgresql i corrent sobre un servidor WildFly

El projecte es centra en la part no coberta de l'aplicació actual que és la gestió de permisos per a usuaris.

Abstract (in English, 250 words or less):

Project for a new version of Intranet for the company Ubilibet S.L. with the aim of providing a system of permissions for users. Made with Java EE, with Postgresql database and running on a WildFly server

The project focuses on the uncovered part of the current application which is user permission management.

Índex

1. Introducció.....	1
1.1 Context i justificació del Treball.....	1
1.2 Objectius del Treball.....	1
1.2.1 Estructura d'Ubilibet.....	1
1.2.2 Identificació i seguretat.....	2
1.3 Enfocament i mètode seguit.....	3
1.4 Planificació del Treball.....	3
1.5 Breu sumari de productes obtinguts.....	4
1.6 Breu descripció dels altres capítols de la memòria.....	4
2. Resta de capítols.....	4
2.1 Arquitectura de l'aplicació.....	4
2.2 Diagrames de classes.....	7
2.2.1 Seguretat.....	7
2.2.2 Dades.....	8
2.3 Casos d'ús.....	9
2.3.1 Identificació d'usuari.....	9
2.3.2 Control d'usuaris.....	10
2.3.3 Manteniment d'usuari.....	10
2.3.3.1 Dades generals.....	11
2.3.3.2 Permisos.....	12
2.3.4 Esborrat.....	13
2.4 Viabilitat.....	13
3. Conclusions.....	14
4. Glossari.....	16
5. Bibliografia.....	16
6. Agraïments.....	16

Llista de figures

Il·lustració 1: Planificació.....	4
Il·lustració 2: Capes de l'aplicació.....	6
Il·lustració 3: Diagrama de classes: Seguretat.....	7
Il·lustració 4: Diagrama de classes: Dades.....	8
Il·lustració 5: Casos d'ús.....	9
Il·lustració 6: Identificació d'usuari.....	9
Il·lustració 7: Control d'usuaris.....	10
Il·lustració 8: manteniment d'usuari, dades generals.....	11
Il·lustració 9: Manteniment d'usuari permisos.....	12
Il·lustració 10: Esborrar usuari.....	13

1. Introducció

1.1 Context i justificació del Treball

L'actual intranet d'Ubilibet tot i tenir un sistema d'identificació d'usuaris no gestiona la possibilitat de limitar l'accés d'aquests a determinades opcions, d'altra banda el fornt-end està fet sols en castellà.

Aquest problema durant tot aquest temps no ha estat tracta perquè el nombre d'usuaris era petit i de confiança, això està canviant perquè Ubilibet s'ha integrat en un grup Europeu i es preveu l'increment d'usuaris i la seva possible implantació en altres països (Portugal, Itàlia i França)

D'altra banda l'extranet per a clients sols permet a aquest veure la informació i sovint se'ns ha demanat poder fer determinades operacions ells mateixos.

La intranet està desenvolupada en PHP i te més de quinze anys i ha passat per les mans de, com a mínim cinc programadors sense connexió entre ells.

l'extranet està realitzada en Python.

1.2 Objectius del Treball

Disposar d'una eina de treball que serveixi tant pels usuaris interns com externs a l'hora que es gestiona quines operacions poden fer, quina informació veuen i en quin idioma treballen.

1.2.1 Estructura d'Ubilibet

L'estructura de l'empresa es compon dels següents departaments:

- Area financera: tindrà accés a la generació de comandes i edició de factures, edició de clients i tarifes.
- Area de consultoria: poden generar pressupostos i clients així com l'edició de descomptes, han de poder veure totes les operacions dels client però no haurien de poder crear-les o cancel·lar-les.

- Area operativa: O producció, genera les peticions als diferents proveïdors per al registre, trasllat, modificació i cancel·lació de noms de domini i altres serveis d'operativa similar (Certificats SSL, TMCH, monitorització, ...), no necessiten tenir accés a descomptes ni preus.
- Area tècnica: realitza el desenvolupament d'aplicacions i manteniment tant del sistema com del programari i maquinari informàtic. Per raons obvies ha de tenir accés a tot.
- Direcció: tot i que teoria el cap d'àrea pertany a l'àrea indicada es pot considerar que caldrà incloure aquest departament, lògicament hauran de tenir accés a tot, tot i que no sempre tindran la possibilitat de realitzar operacions, també tindran accés a alguns informes que en aquest moment o tothom té accés o no estan inclosos.

1.2.2 Identificació i seguretat

L'objectiu principal del TFG és crear tota l'estructura d'usuaris i permisos per a poder accedir a la Intranet, incloent un manteniment d'usuaris que permeti crear-ne, modificar i eliminar-lo, així com assignar-hi permisos per a les diferents funcions.

Els usuaris poden pertànyer a una d'aquestes àrees o bé ser clients autoritzats a accedir-hi, lògicament sols a les seves dades. Un usuari podrà tenir diferents nivells de permís per a cada funció del sistema, la manca de definició de permís implica la impossibilitat d'accedir a la funció.

Cada usuari tindrà assignat un idioma per la interfície, als informes es proposarà en alguns casos l'idioma de l'usuari, en d'altres l'idioma del client tanmateix es podrà seleccionar l'idioma a utilitzar.

Es considera una funció tant una opció del menú com el tractament d'un camp determinat dins un manteniment o un tipus de dades, per exemple alguns noms de domini, amb un atribut determinat sols podran ser modificats per un administrador: imaginem els DNS de uoc.edu, si es canvien sense deixarien de funcionar pràcticament tots els serveis de la universitat, però canviar els DNS de uoc.cat o uoc.es sols deshabilita un redireccionament.

Els diferents nivells de permís haurien de ser els següents:

- Usuari sols lectura limitat: es tractarà d'un usuari generalment associat amb un client o grup de clients les consultes es limitaran a les dades corresponents als clients associats. Sols podrà veure les dades i generar informes o documents PDF (per exemple les seves factures).
- Usuari edició limitat: similar al anterior, podrà realitzar algunes operacions, per exemple crear peticions de registre o expiració de noms de domini.
- Usuari sols lectura: tindrà accés a les dades però no podrà modificar-les, per exemple un membre de Consultancy Area podrà veure la factura d'un client però no podrà crear una factura rectificativa a partir de la mateixa, podrà veure les peticions a proveïdors però no crear-ne.
- Usuari edició: tindrà accés a les dades i podrà modificar-les.
- Administrador: tindrà accés a les dades restringides, inclosa la gestió d'usuaris, en ocasions se li reclamarà la introducció d'una contrasenya per a poder realitzar l'operació.

1.3 Enfocament i mètode seguit

La primera opció hauria estat incloure el control de permisos en l'actual intranet, la seva implementació seria prou complexa degut a que ja d'entrada no podem parlar de programació estructurada ni per suposat de programació orientada a objectes o qualsevol altre paradigma o patró de programació.

La possibilitat d'incloure diferents idiomes per als usuaris a l'actual intranet està descartada atès que tots els literals i les cadenes d'html estan escampats dins del codi.

En conseqüència la solució passa per implementar una nova intranet. Donada la meua experiència amb Java i Hibernate he preferit generar la implementació de la nova intranet en amb JavaEE tot i reconèixer que la meua experiència amb JSF i EJB és limitada. Això implica que treballarem amb una arquitectura per capes on la capa d'integració la realitzarem amb JPA, la capa de negoci amb EJB i la capa de presentació amb JSF.

1.4 Planificació del Treball

Malgrat a la PAC 1 vaig preveure una segona fase on es gestionés el contactes de facturació (clients en l'argot d'Ubilibet) pensant que la corba d'aprenentatge em permetria si més no iniciar-ho la realitat és que no he pogut ni tan sols plantejar-me iniciar-ho per la qual cosa he reformulat la planificació intentant ser més realista.

WBS	Nom	Comença	Acaba	Feina
1	PAC 1	Feb 20	Mar 9	7d 3h
1.1	Documentació	Feb 20	Feb 29	3d 5h
1.2	Eines	Feb 29	Mar 9	3d 5h
2	Lliurament PAC1	Mar 6	Mar 6	
3	PAC 2	Mar 6	Apr 8	13d 6h
3.1	Disseny casos d'ús	Mar 6	Mar 19	5d 4h
3.2	Disseny pantalles	Mar 19	Mar 28	3d 5h
3.3	Disseny Base de dades	Mar 28	Mar 30	5h
3.4	Disseny classes	Mar 30	Apr 8	3d 5h
4	Lliurament PAC2	Apr 10	Apr 10	
5	PAC 3	Apr 10	May 26	24d 4h
5.1	Aplicació base	Apr 10	Apr 11	5h
5.2	Classes JPA	Apr 11	Apr 16	1d 6h
5.3	Classe EJB seguretat	Apr 16	Apr 20	1d 6h
5.4	Cas d'ús identificació	Apr 20	Apr 25	1d 6h
5.5	Menú aplicació	Apr 25	Apr 27	1d
5.6	Test de l'aplicació	Apr 27	Apr 28	2h
5.7	Gestio Idiomes	Apr 28	Apr 29	2h
5.8	Cas d'us control usuaris	Apr 29	May 5	2d 1h
5.8.1	Visualització	Apr 29	May 4	1d 6h
5.8.2	Esborrat	May 4	May 5	2h
5.9	Cas d'ús manteniment usuaris	May 5	May 26	14d 1h
5.9.1	Creació	May 5	May 18	5d 4h
5.9.2	Assignació permisos	May 5	May 23	7d 3h
5.9.3	Polít de detalls (canvi de contrasenya, introducció email, ...)	May 23	May 26	1d
6	Test aplicació i proves instal·lació	May 26	May 29	1d
7	Lliurament PAC3	May 29	May 29	

II-lustració 1: Planificació

1.5 Breu sumari de productes obtinguts

El producte obtingut és la base per a una aplicació on es pot configurar d'una manera individual els permisos per a cada usuari, la configuració d'un nou idioma és prou senzilla com per poder incloure'l sols amb la generació i incorporació d'un nou arxiu a l'aplicació.

1.6 Breu descripció dels altres capítols de la memòria

1. Arquitectura de l'aplicació
Definició dels requeriments
2. Diagrama de classes
Descripció de la base de dades
3. Casos d'ús
Cadascun dels casos creats en l'aplicació

2. Resta de capítols

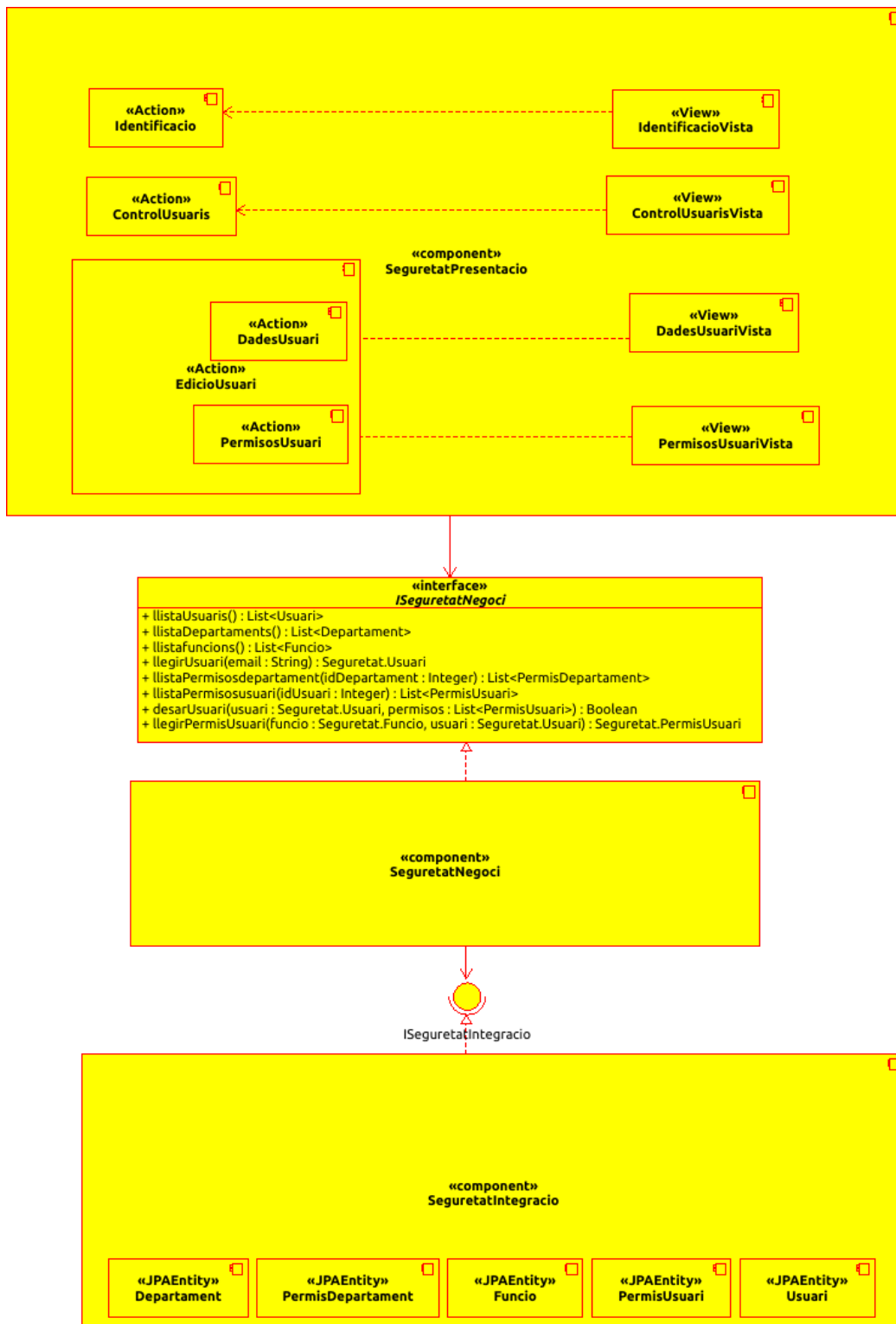
2.1 Arquitectura de l'aplicació

Fem servir una estructura per capes on hi ha una capa de presentació JSF que s'encarrega de la presentació i l'interfase amb els usuaris, una capa de negoci EJB i una capa d'integració JPA

La capa de negoci gestiona tots els accessos a la capa d'integració i assegura la coherència de les dades.

La capa de presentació JSF correspon al patró MVC on les vistes es realitzen amb arxius xhtml, els controladors són Managed Beans i es creen classes model que sovint tenen una relació directa amb les classes JPA.

Des de la capa de presentació mai no es modifiquen dades de classes JPA, per realitzar les operacions es crida sempre als mètodes adjacents de la capa de negoci que s'encarrega de llegir i modificar les dades a la capa d'integració.



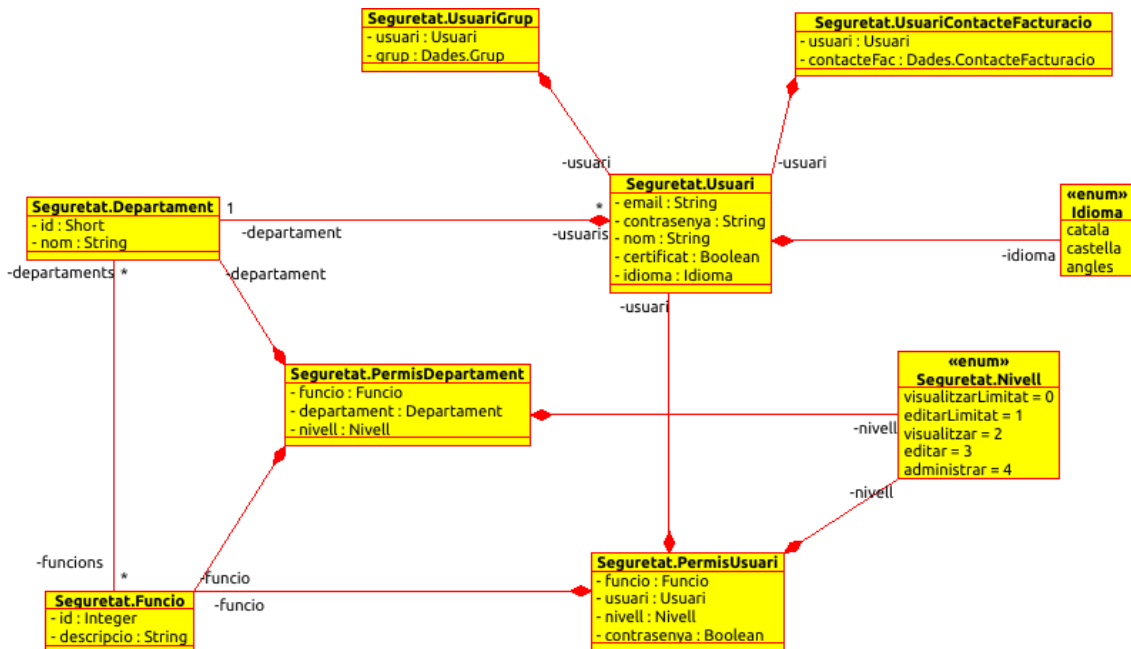
Il·lustració 2: Capes de l'aplicació

2.2 Diagrames de classes

Donat que en un principi tenia pensat realitzar una segona fase on gestionar els contactes de facturació a la capa d'integració es van incloure les classes corresponents en aquest futur mòdul

2.2.1 Seguretat

En aquest diagrama mostrem les classes que afecten a la seguretat de l'aplicació.

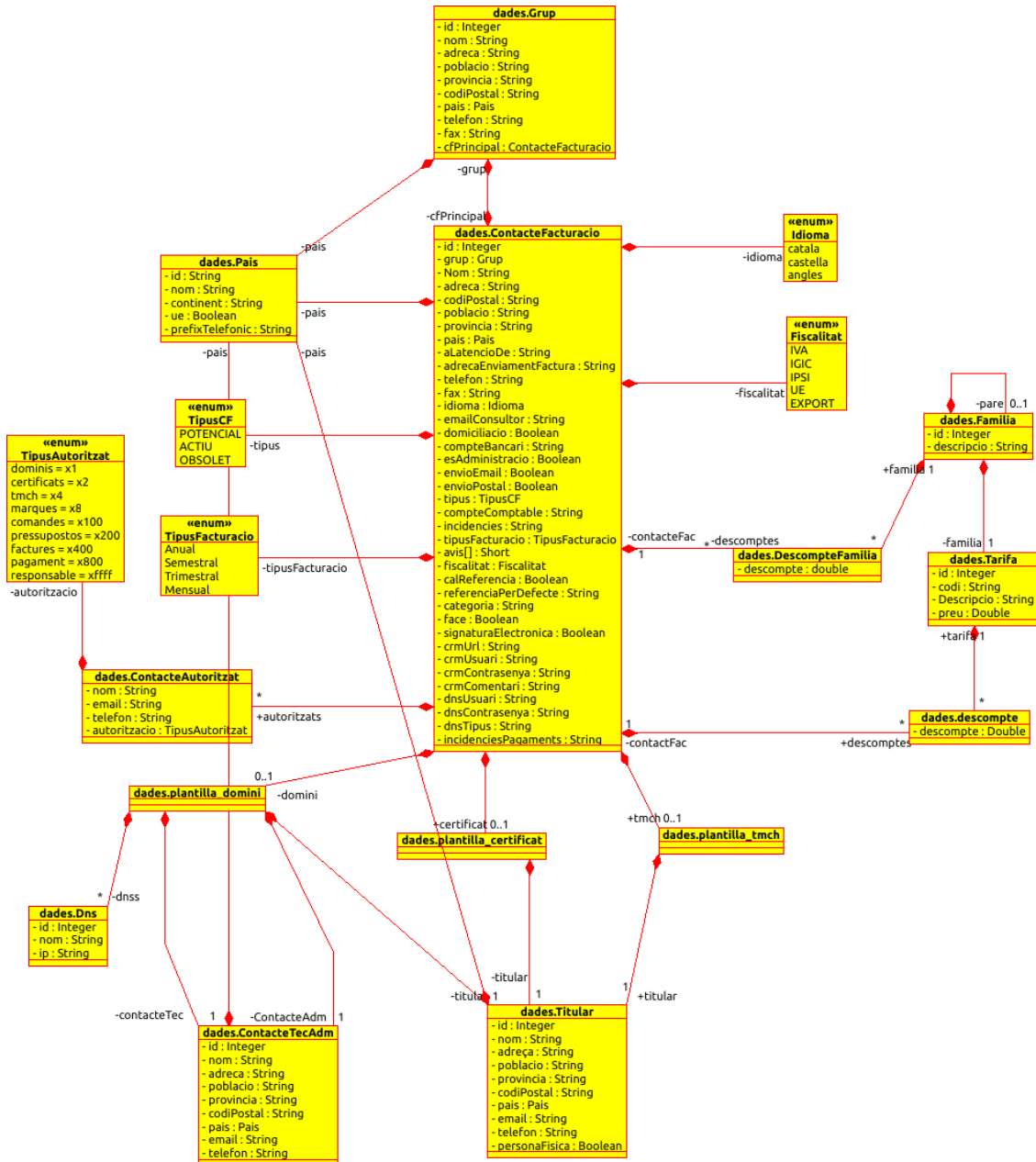


Il·lustració 3: Diagrama de classes: Seguretat

Un aspecte a tenir en compte és que el correu electrònic de l'usuari és únic ja que el farem servir a la identificació.

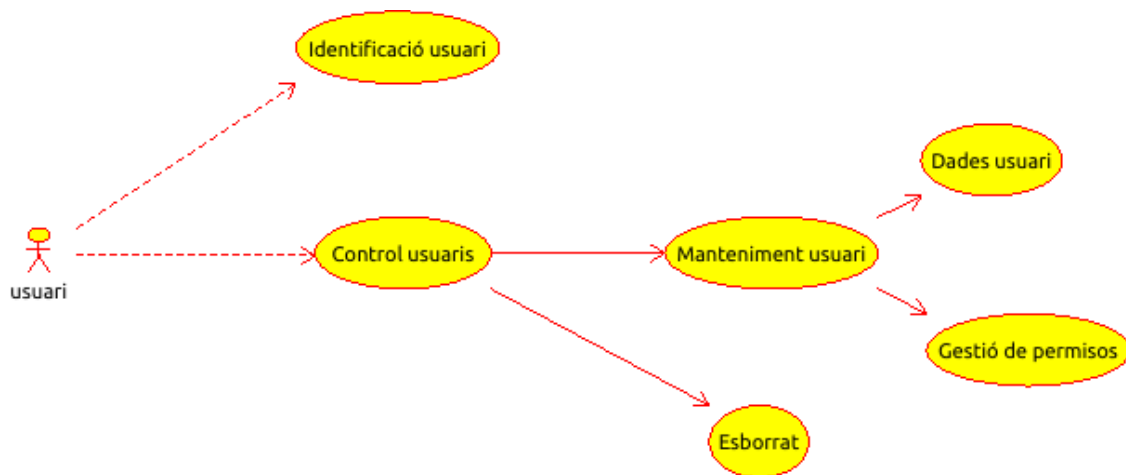
2.2.2 Dades

En aquest diagrama mostrem les classes corresponents a dades. Incloc sols les dades que afecten al projecte en aquesta fase, en fases posteriors les classes que els afecte (dominis, certificats, monitoritzats, pressupostos, comandes, factures ...)



Il·lustració 4: Diagrama de classes: Dades

2.3 Casos d'ús



Il·lustració 5: Casos d'ús

2.3.1 Identificació d'usuari

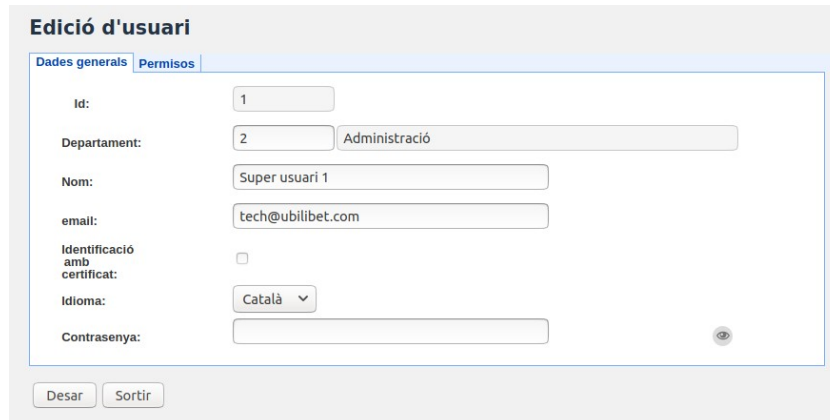
L'usuari introdueix el seu correu electrònic i la seva contrasenya, en prémer el botó «login» es comproven les credencials i en cas positiu es produeix la identificació tenint accés a partir d'aquest moment a les opcions autoritzades, al Mteix temps s'activa l'idioma de l'usuari en tota l'aplicació

Captura de pantalla d'un formulari de login. A la part superior hi ha els enllaços 'Database' i 'Domains'. El títol principal és 'Login'. Hi ha dos camps d'entrada: 'User:' i 'Pasword:'. El camp 'Pasword:' té un icona per alternar la visibilitat de la contrasenya. A sota dels camps hi ha un botó de login amb un símbol d'usuari i una clau.

Il·lustració 6: Identificació d'usuari

2.3.3.1 Dades generals

Es s'introdueixen nom, idioma (llista tancada), correu electrònic, contrasenya i departament al que pertany l'usuari.



The screenshot shows a web form titled "Edició d'usuari" with two tabs: "Dades generals" (selected) and "Permisos". The form contains the following fields and controls:

- Id:** A text input field containing the number "1".
- Departament:** A text input field containing "2" and a dropdown menu showing "Administració".
- Nom:** A text input field containing "Super usuari 1".
- email:** A text input field containing "tech@ubilibet.com".
- Identificació amb certificat:** A checkbox that is currently unchecked.
- Idioma:** A dropdown menu showing "Català".
- Contrasenya:** A text input field for a password, with a small eye icon to its right for toggling visibility.

At the bottom of the form, there are two buttons: "Desar" (Save) and "Sortir" (Exit).

Il·lustració 8: manteniment d'usuari, dades generals

Al indicar l'identificador del departament el managed bean fa una consulta a la capa de negoci, si existeix es canvia la llista actual de permisos per la que tingui definida el departament. En cas que el departament no existeixi es mostra un missatge advertint que no es permetrà desar la informació.

Per evitar pèrdues d'informació si l'identificador és igual a l'anterior no es realitza cap canvi.

El camp contrasenya s'utilitza per assignar una nova contrasenya. l'antiga no es mostra i sols si s'ha introduït informació vàlida es canvia.

En aquest moment es permet una contrasenya conformada amb un mínim de deu caràcters, com a mínim una majúscula, una minúscula i un dígit.

2.3.3.2 Permisos

Apareix una llista amb les funcions disponibles el seu nivell de permís per departament, el nivell de permís de l'usuari i si necessita contrasenya per executar la funció, a cada funció se li pot modificar el nivell de seguretat i si cal que torni a introduir la contrasenya.

Funció	Nivell	Cal contrasenya	Esborrar
Edició Usuaris	Administrar	<input type="checkbox"/>	<input type="checkbox"/>
Control Usuaris	Administrar	<input type="checkbox"/>	<input type="checkbox"/>
Control Departaments	Administrar	<input type="checkbox"/>	<input type="checkbox"/>
Edició permisos usuaris i canvi departament	Administrar	<input type="checkbox"/>	<input type="checkbox"/>

Nou Permis: Seleccionar

Desar Sortir

Il·lustració 9: Manteniment d'usuari permisos

Es tracta d'una llista incrementable, usualment la primera vegada, o quan es canviï el departament, s'aplicaran els permisos que tingui configurat el departament.

A la llista «Nivell» es pot canviar entre els valors «visualitzar limitat», «visualitzar», «editar limitat», «editar» i «administrar».

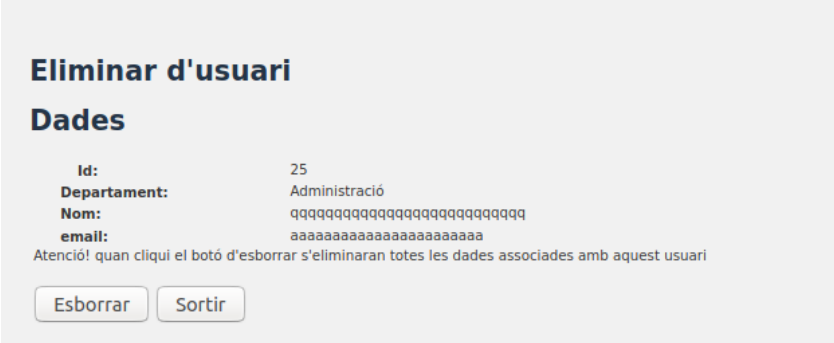
La casella «Cal contrasenya» s'utilitza per indicar que per realitzar la funció cal introduir novament la contrasenya. És possible que aquest control acabi cridant un segon factor de validació a l'estil de Google authenticator.

La casella «Esborrar» maraca el permís per ser esborrat a l'hora de desar la informació. En aquest cas tenia dubtes si fer-ho directe, es a dir eliminar visualment el permís o mantenir-ho en al vista fins desar. Al final vaig optar per la segona opció.

Per afegir un nou permís es fa servir la llista desplegable que apareix al peu, en el moment en que seleccionem un es revisa la llista de permisos per veure si ja el tenim inclòs, en cas que ja estigui el sistema ens avisa i no realitza cap acció, si no hi és afegeix un nou permís a la llista amb la funció seleccionada.

2.3.4 Esborrat

Mostra les dades de l'usuari a eliminar, l'eliminació es confirma prement el botó «Esborrar».



The screenshot shows a dialog box titled "Eliminar d'usuari" (Delete user). Below the title is a section labeled "Dades" (Data) containing the following information:

Id:	25
Departament:	Administració
Nom:	qqqqqqqqqqqqqqqqqqqqqqqqqqqqqq
email:	aaaaaaaaaaaaaaaaaaaaaaaa

Below the data is a warning message: "Atenció! quan cliqui el botó d'esborrar s'eliminaran totes les dades associades amb aquest usuari" (Attention! when clicking the delete button all data associated with this user will be deleted). At the bottom of the dialog are two buttons: "Esborrar" (Delete) and "Sortir" (Cancel).

Il·lustració 10: Esborrar usuari

2.4 Viabilitat

Sense cap gènere de dubte aquesta aplicació tot i ser operativa no gestiona encara la línia de negoci d'Ubilibet, malgrat això dóna una base que permetrà renovar la Intranet actual i poder oferir una extranet als clients on puguin realitzar consultes i operacions.

La viabilitat de la mateixa dependrà en molt de la voluntat de seguir la implementació de la mateixa per part de la direcció d'Ubilibet. Els impediments per seguir amb aquesta implementació són la necessitat de vincular la informació d'Ubilibet al grup, que en aquest moment és prioritari, la constant automatització de serveis que sovint ens obliga a aparcar temes i, vinculat amb els dos primers, els recursos limitats del departament pel que fa a personal, per aclarir, i sense cap ànim de personalisme, ans al contrari, el departament tècnic d'Ubilibet sóc jo.

3. Conclusions

Al començar amb una aplicació d'exemple de Jboss he treballat també amb CDI, el primer gran problema per a mi ha estat entendre com operen JSF i CDI, atès que la meva experiència en Java està molt més vinculada a aplicacions de backend, això implica que determinats conceptes com ara els àmbits (scopes) s'em resistissin. Tot i això crec que començo a entendre'ls.

De camí he après a gestionar xhtml, tot i que l'actual Intranet el fa servir tot el que feia era copiar i enganxar; sí, entenia algunes coses però vincular html amb JSF o amb PHP és prou diferent.

La programació de la capa de presentació ha estat la part més complicada, no tanta pel patró MVC en si mateix sinó per la gestió de tags de CDI/JSF.

El primer problema va ser entendre els àmbits `SessionScope`, `ViewScope` i `RequestScope`, De fet el descobriment de `ViewScope` és el que em va permetre trobar la solució que cercava. En el calaix ha quedat el `ConversationScoped` que, tot i poder ser també una solució no en vaig trobar un exemple que no em fes passar per tres vistes diferents quan el que jo volia era gestionar una de sola.

Resumint un managed bean `SessionScoped` es manté durant tota la sessió, un `ViewScoped` es perd al sortir de la vista i un `RequestScoped` es crea amb la vist, es destrueix al acabar de mostrar-la i es torna a crear amb cada submit.

El següent problema el vaig tenir amb l'actualització de les vistes que no sempre van funcionar com jo volia. En el cas de departament hauria volgut que al introduir un id inexistent pogués actualitzar el camp de la visat amb el valor previ, no vaig torbar la manera d'actualitzar-ho correctament, cosa que sí passava amb el nom de del departament. Queda per un futur aconseguir-ho.

Per a la validació del correu electrònic vaig provar `validatorRegex` però no em va acabar de convèncer i vaig optar per comprovar-ho abans de desar.

Hauria volgut afegir un nou cas d'ús que era la selecció del departament en una vista apart tot i que la tenia bastant avançada no acabava de tenir clar com retornar el resultat al managed bean d'origen i, calendari en mà, vaig optar per no incloure una funcionalitat a mitges.

La meva experiència amb Hibernate m'ha servit per entendre JPA sense gaire esforç, la part que més m'ha costat ha estat el delegar en JPA la gestió de les transaccions, he de dir però que encara no tinc clar cedir-les, en alguns casos el fet de gestionar-les jo m'ha permès establir una

transacció coordinada de vàries bases de dades (al comptabilitzar factures, base de dades de gestió i base de dades de comptabilitat)

Respecte a EJB he de reconèixer que no l'he explotat com podria fer-ho, és molt possible que una tercera versió sigui fer que la capa de negoci sigui distribuïda.

Pel que fa als objectius plantejats al principi, a la PAC 1, puc dir sense equivocar-me que vaig estar massa optimista, crec que aquest error per part meva no va ser tant per creure'm capaç de fer-ho sinó per pensar que la primera fase donava un resultat en principi massa senzill.

Em vaig equivocar en els objectius però tenia raó en la imatge de la feina, sé tota la feina que m'ha portat i el resultat aconseguit no reflecteix al meu entendre aquesta feina.

Tot i així de tot el que volia fer en la primera fase, gestió d'usuaris, considero que he cobert un 90% dels objectius, sols m'ha quedat la selecció de departament a partir d'una llista consultable que, si mireu el codi està embastada.

D'altra banda un dels darrers canvis va ser mostrar els missatges si es produïa algun problema abans de desar la informació, no em vaig adonar i al fer el vídeo de demostració vaig veure que apareixien missatges no desitjats durant l'execució.

He provat de seguir la planificació però, atès que sovint em quedava bloquejat sovint vaig optar per avançar altres aspectes per poder tenir, si més no, feina avançada.

Com deia a l'apartat de viabilitat si els meus caps d'Ubilibet s'animen tirarem endavant aquesta nova versió de la Intranet, potser amb un replantejament més progressiu, acabant tot el mòdul de seguretat abans de passar a la resta de mòduls administració, comercial, dominis, certificats, ... Estic segur que pot ser una gran eina.

Segurament aquesta és una de les memòries més curtes que s'han fet però tinc tendència a resumir i donar per sabudes explicacions que potser hauria d'incloure.

Segurament l'aplicació ha quedat massa bàsica i segurament hauria hagut d'inventar-me una altra per fer quelcom més vistós. Tot i així el resultat és un bon punt de partida per al que vull fer, espero que també sigui suficient per a aquest treball de final de grau.

4. Glossari

Departament: Engloba un grup d'usuaris que realitzen feines similars.

Usuari: persona amb autorització per operar a la intranet, principalment personal d'Ubilibet i clients autoritzats.

Funció: Qualsevol opció de menú o operació a la que se li vulgui aplicar algun tipus de restricció.

Permís: Atribut que se li assigna a un usuari respecte a una funció.

Contacte de facturació: Històricament els clients han estat anomenats com contactes de facturació, atès que la Intranet es va fer per gestionar els noms de domini, aquests tenien els següents «contactes»: propietari o titular, contacte administratiu, contacte tècnic i contacte de facturació.

5. Bibliografia

Aquesta és la llista d'enllaços dels que m'he nodrit, la majoria han estat consultats constantment i per diferents conceptes:

1. MkYong (<https://mkyong.com/jsf2>)
2. FalconMasters (<http://www.falconmasters.com/>) pel menú desplegable
3. Stack Overflow (<https://stackoverflow.com/>)
4. Jboss (<https://docs.wildfly.org>)
5. Icon Archive (<http://www.iconarchive.com/>) per les icones per als botons

6. Agraïments

Vull agrair la paciència que han tingut:

La meva família per les hores dedicades no sols a aquest TFG sinó també als més de deu anys que he dedicat al grau.

El meu consultor l'Albert Grau Perisé per la seva paciència amb les meves consultes i emprenyades quan en quedava bloquejat.

Als meus caps d'Ubilibet que m'han animat a seguir endavant i m'han facilitat les coses per poder realitzar aquest treball.