

2015

# TFC: PORTAL PER EMPRESES DE HOSTING

Memòria

Autor: Eduard Gubert Bril·li

ETIS

Consultor: José Juan Rodriguez

12 de Gener de 2015



## RESUM

Aquest treball té per objectiu l'anàlisi, disseny i implementació d'una aplicació mitjançant tecnologia J2EE. Més concretament, la creació d'un portal destinat a empreses dedicades al registre de dominis i serveis d'allotjament a internet.

En la part inicial del document s'exposa el punt de partida del projecte així com els objectius generals i específics que s'esperen aconseguir. Aquests, inclouen des de l'aprenentatge de la tecnologia fins a la implementació final de l'aplicació, que en aquest cas, es limita a només una part de l'abast total contingut en l'anàlisi.

L'anàlisi inclou l'estudi de casos d'ús de l'aplicació, el diagrama UML de classes i els esquemes de les diferents pantalles. Addicionalment, es fa una breu introducció al patró Model-Vista-Controlador, la arquitectura base que utilitza el portal.

Posteriorment, es fa una exposició detallada del producte obtingut, incidint especialment en els aspectes funcionals, els canvis i decisions preses durant la fase d'implementació així com la proposta de futures funcionalitats o millores.

En la part final del treball hi trobem les instruccions necessàries per poder instal·lar l'aplicació. Aquestes inclouen l'estructura de la base de dades, les credencials dels usuaris de prova i els paràmetres recomanats d'execució.

**Àrea del TFC:** J2EE

**Paraules clau:** j2ee, spring, tomcat, tiles, hibernate, mvc, hosting, portal, domini, allotjament

## ÍNDEX

RESUM.....	1
ÍNDEX.....	2
1. JUSTIFICACIÓ DEL PROJECTE .....	5
2. OBJECTIUS .....	5
2.1 OBJECTIUS GENERALS .....	5
2.2 OBJECTIUS ESPECÍFICS .....	5
3. DESCRIPCIÓ DEL PROJECTE.....	7
4. PLANIFICACIÓ .....	8
4.1 FITES .....	8
5. ANÀLISI.....	9
5.1 ACTORS.....	9
5.2 DIAGRAMA DE CASOS D'ÚS .....	10
5.3 DESCRIPCIÓ TEXTUAL DELS CASOS D'ÚS MÉS IMPORTANTS.....	11
5.4 ESQUEMA DE PANTALLES PRINCIPALS .....	27
5.4.1 PORTADA .....	27
5.4.2 REGISTRE D'USUARI .....	27
5.4.3 INICIAR SESSIÓ .....	28
5.4.4 ÀREA PRIVADA – CLIENT .....	28
5.4.5 ÀREA PRIVADA TREBALLADOR .....	29
5.4.6 ÀREA PRIVADA ADMINISTRADOR.....	30
5.4.7 CATÀLEG DE PRODUCTES .....	31
5.4.8 CATÀLEG DE PRODUCTES - DOMINIS .....	31
5.4.9 CATÀLEG DE PRODUCTES – ALLOTJAMENTS.....	32
5.5 DIAGRAMA UML DE CLASSES .....	33
5.6 ARQUITECTURA .....	34
5.6.1 EL MODEL .....	34
5.6.2 LA VISTA.....	34
5.6.3 EL CONTROLADOR .....	34

5.6.4	SPRING MVC.....	35
6.	IMPLEMENTACIÓ I PRODUCTES OBTINGUTS.....	36
6.1	COMPONENTS DE L'APLICACIÓ.....	36
6.1.1	LA VISTA – DISSENY DE LA INTERFÍCIE D'USUARI .....	36
6.1.2	MÒDUL D'USUARIS .....	39
6.1.3	PERSISTÈNCIA DE DADES.....	40
6.1.4	CONTROLADORS I LA CAPA DE NEGOCI.....	41
6.1.5	CONSULTA DE DISPONIBILITAT DE DOMINIS .....	42
6.1.6	TESTING I LOGGING .....	44
6.1.7	LOCALITZACIÓ DE MISSATGES .....	45
7.	MODIFICACIONS DEL PLANTEJAMENT INICIAL.....	46
7.1	MODEL UML .....	46
7.2	ABAST DEL PROJECTE .....	46
8.	CONCLUSIONS.....	47
9.	GLOSSARI .....	47
10.	BIBLIOGRAFIA.....	48
11.	ANNEX 1 – INSTAL·LACIÓ DE L'APLICACIÓ .....	49
12.	ANNEX 2 – SCRIPT CREACIÓ DB I DADES DE MOSTRA .....	50



## 1. JUSTIFICACIÓ DEL PROJECTE

La tecnologia J2EE és un dels estàndards més estesos en el món empresarial, i aquest ha estat el motiu principal pel qual he escollit aquesta àrea per desenvolupar el meu treball final de carrera.

Pel que fa a la temàtica de l'aplicació, actualment treballo al departament d' ISP d'una empresa de serveis informàtics i m'ha semblat molt interessant la creació d'una aplicació d'aquestes característiques, donat que tracta una temàtica on ja tinc experiència i això m'ha permès centrar-me més en l'aprenentatge de la tecnologia J2EE, en la qual no tenia cap tipus d'experiència més enllà de la programació java bàsica, i no tant en els requisits o les funcionalitats que ha d'implementar l'aplicació. A més, crec que en un futur, pot esdevenir una aplicació de producció extremadament útil.

## 2. OBJECTIUS

### 2.1 OBJECTIUS GENERALS

Com ja hem dit anteriorment, l'objectiu principal del projecte és aprofundir en la tecnologia J2EE, i desenvolupar una aplicació web sota l'arquitectura MVC amb persistència de dades.

### 2.2 OBJECTIUS ESPECÍFICS

A més, com a objectiu secundari, intento prioritzar l'ús de tecnologies vigents o en proliferació en escenaris corporatius reals.

Aprendre i implementar les següents eines o tecnologies:

- L'IDE Eclipse



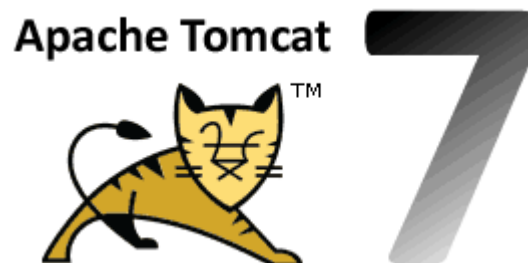
- El framework Spring



- L'ORM Hibernate



- Apache Tomcat



- Mysql



- Subversion



Durant tot el transcurs del projecte, el codi font estarà allotjat en un repositori públic de control de versions a google, i s'aniran pujant els canvis de forma regular, facilitant d'aquesta manera la creació de la memòria del projecte un cop finalitzat.

### 3. DESCRIPCIÓ DEL PROJECTE

L'aplicació que he escollit per desenvolupar com a TFC consisteix en un portal de gestió i automatització d'una empresa ISP, inspirada en aplicacions de similar funcionalitat ja existents en el mercat com ara WHMCS: <http://www.whmcs.com/>

Concretament, es tracta d'implementar un portal que integri les següents funcionalitats:

Per la part del client (FrontEnd):

- Compra i gestió de dominis
- Compra i gestió d'allotjaments
- Renovació de productes
- Wiki amb manuals o documentació útil

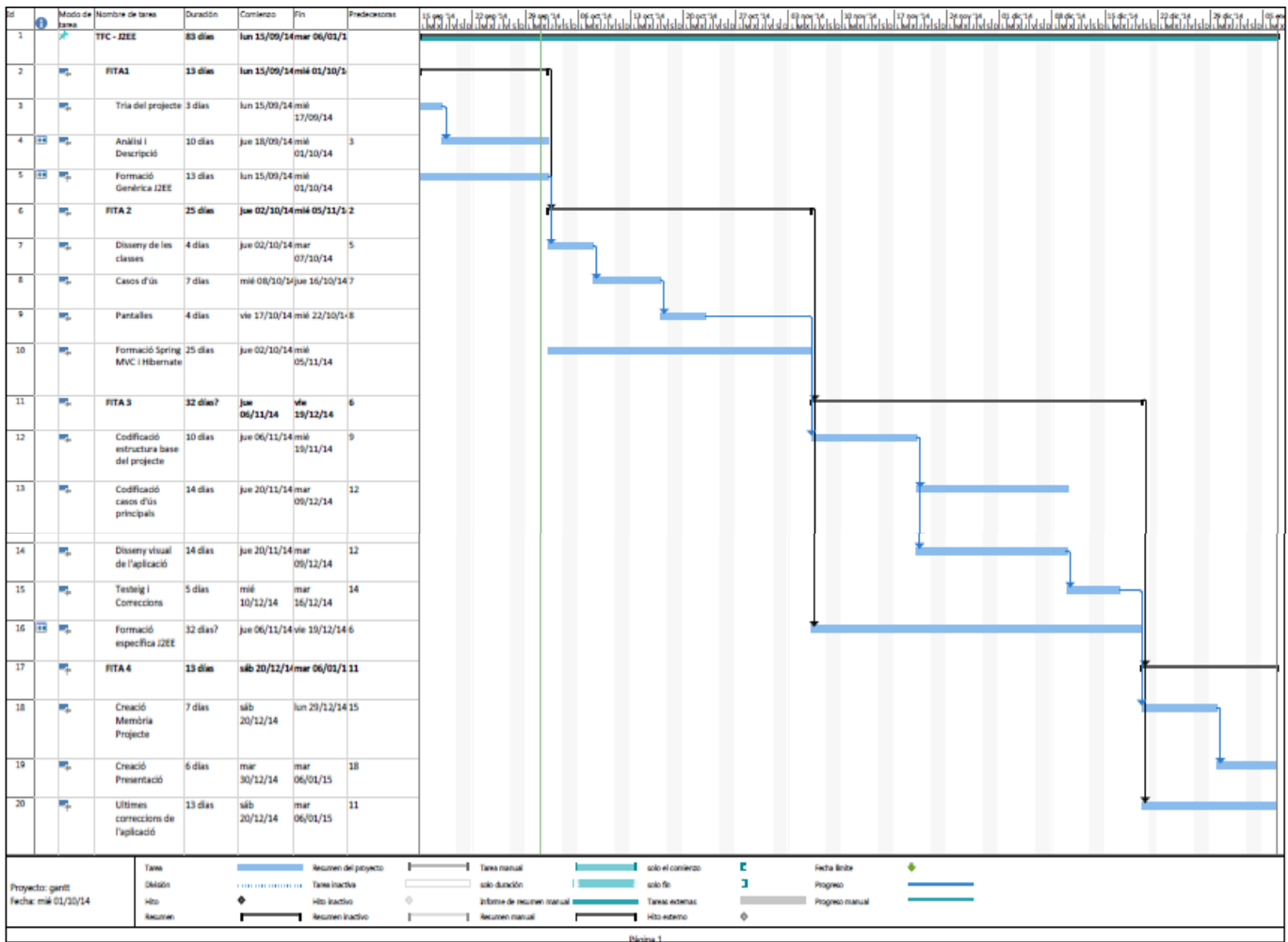
Per la part de l'empresa (BackEnd)

- Gestió d'usuaris
- Integració amb el programari Plesk (auto-aprovisionament d'allotjaments)
- Integració amb un registrador de dominis (OVH)
- Opcions globals de configuració
- Gestió dels productes (preus, característiques, correspondència de productes amb plesk, etc..)

La idea principal és que l'aplicació es compozi de mòduls, i cada mòdul implementi un dels requisits prèviament exposats i, d'aquesta manera, sigui fàcilment ampliable amb noves funcions.



## 4. PLANIFICACIÓ



### 4.1 FITES

#### FITA 1 - 1/10/2014: Pla de treball

Tria, Anàlisi i Descripció del Projecte

#### FITA 2 – 5/11/2014: Especificació de l'aplicació

Disseny de les classes, casos d'ús, interfície de l'usuari  
 Creació de la base de l'aplicació: Obtenir una base funcional de programari sobre la qual poder començar a treballar, i afegir funcionalitats en forma de mòduls.

#### FITA 3 – 19/12/2014: Desenvolupament del projecte

Obtenir una versió funcional de l'aplicació, codificant les classes, pantalles i les funcionalitats acordades d'acord amb l'especificació del projecte.

**FITA 4 – 06/01/2014: Memòria i presentació**

Redacció de la memòria del projecte i creació de la seva presentació.

## 5. ANÀLISI

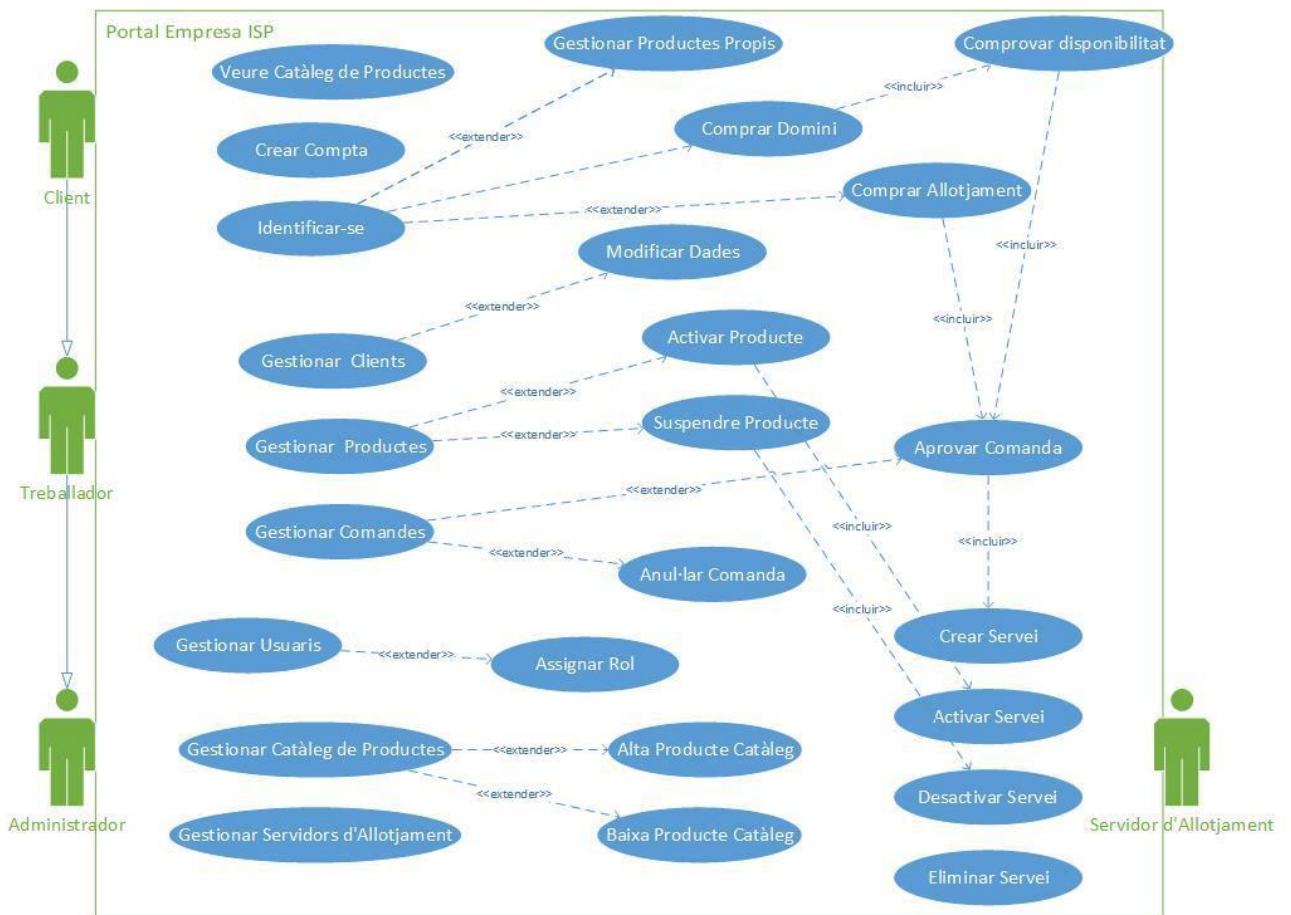
A continuació analitzarem el projecte proposat, descrivint els actors i els casos d'ús més importants.

### 5.1 ACTORS

Els actors que intervenen en el portal ISP són:

- **Client:** Tal i com el seu propi nom indica, els clients son els usuaris finals que podran contractar i gestionar els seus serveis.
- **Treballador:** El treballador és l'usuari donarà suport als clients, tindrà accés a les dades dels clients que tinguin serveis contractats a l'empresa, i seran els responsables de validar les comandes. Addicionalment , també poden modificar les dades personals dels clients així com, suspendre o activar els seus productes.
- **Administrador:** És l'usuari més privilegiat del portal, la seva tasca principal és la de mantenir el catàleg de productes , assignar el rol de treballador als usuaris ( per defecte tots els nous usuaris es consideren clients) . A més, l'administrador serà el responsable de totes les configuracions possibles del portal: Paràmetres de comunicació amb els servidors d'allotjament, gestió dels continguts dinàmics, etc..
- **Servidor d'allotjament:** Tot i que el servidor d'allotjament no és cap persona, és un element necessari per tal de donar servei als clients. Essencialment, aquest servidor és qui allotja els dominis i espais web que els clients contracten, i per tant, qui rep dels peticions del portal ISP per tal de crear o suspendre allotjaments.

5.2 DIAGRAMA DE CASOS D'ÚS



## 5.3 DESCRIPCIÓ TEXTUAL DELS CASOS D'ÚS MÉS IMPORTANTS

**Cas d'ús:** Veure catàleg de Productes

**Actor Principal:** Client

**Àmbit:** Portal ISP

**Nivell d'objectiu:** Usuari

**Stakeholders i interessos:**

Usuari: Vol veure els diferents productes que ofereix l'empresa

**Precondició: Cap**

**Garanties mínimes:** Hi ha d'haver productes disponibles al catàleg

**Garanties en cas d'èxit:** L'usuari pot veure els diferents productes que s'ofereixen

**Escenari principal d'èxit:**

1. L'usuari indica que vol veure el catàleg de productes
2. El sistema li mostra una llista de categories disponibles
3. L'usuari selecciona la categoria desitjada
4. El sistema li mostra tots els productes que pertanyen a la categoria.

**Extensions:**

3a. La categoria que ha seleccionat l'usuari no té productes disponibles, el sistema li mostra el missatge "Cap producte disponible"

**Cas d'ús:** Crear Compta

**Actor Principal:** Client

**Àmbit:** Portal ISP

**Nivell d'objectiu:** Usuari

**Stakeholders i interessos:**

Usuari: Vol donar-se d'alta al portal per poder adquirir productes

Empresa: Necessita registrar les dades del clients per poder oferir serveis i facturar-los.

**Precondició:** Cap

**Garanties mínimes:** El sistema guarda a la base de dades, les dades introduïdes per l'usuari.

**Garanties en cas d'èxit:** L'usuari podrà accedir després a una àrea privada mitjançant usuari i paraula clau.

**Escenari principal d'èxit:**

1. L'usuari escull la opció de crear compta
2. El sistema mostra un formulari que requereix DNI/NIF, Nom, Adreça, Email i telèfon.
3. L'usuari introdueix les dades sol·licitades
4. El sistema valida les dades i les enregistra a la base de dades.

**Extensions:**

3a. L'usuari introdueix dades incoherents

3a1. El sistema detecta una incoherència en les dades introduïdes, notifica a l'usuari i torna a mostrar el formulari

**Cas d'ús:** Identificar-se

**Actor Principal:** Client

**Àmbit:** Portal ISP

**Nivell d'objectiu:** Usuari

**Stakeholders i interessos:**

Usuari: Vol accedir a la seva àrea privada

Empresa: Vol realitzar el seguiment dels usuaris que s'identifiquen al portal.

**Precondició:** Ha de tenir una compta creada en el sistema.

**Garanties mínimes:** El sistema enregistra l'intent d'identificació

**Garanties en cas d'èxit:** L'usuari accedeix a la seva àrea privada

**Escenari principal d'èxit:**

1. L'usuari indica al sistema que vol iniciar sessió.
2. El sistema li sol·licita nom d'usuari i paraula de pas
3. L'usuari introdueix les seves credencials
4. El sistema verifica que les credencials són correctes i li mostra l'àrea privada.

**Extensions:**

3a. L'usuari introdueix credencials incorrectes

3a1. El sistema li indica que les credencials no són vàlides i torna a mostrar el formulari d'inici de sessió.

**Cas d'ús:** Comprar Domini

**Actor Principal:** Client

**Àmbit:** Portal ISP

**Nivell d'objectiu:** Usuari

**Stakeholders i interessos:**

Usuari: Vol comprar un domini

**Precondició:** L'usuari s'ha d'haver identificat

**Garanties mínimes:** El sistema enregistra la petició de l'usuari

**Garanties en cas d'èxit:** El sistema enregistra la comanda del domini

**Escenari principal d'èxit:**

1. L'usuari indica que el nom de domini que desitja comprar
2. El sistema comprova si està lliure i mostra a l'usuari la opció de compra
3. L'usuari confirma la compra del domini
4. El sistema processa la petició i enregistra la comanda

**Extensions:**

1a. L'usuari indica un nom de domini no disponible.

1a1. El sistema li mostra el missatge "Domini no disponible" juntament amb les dades de registre del domini sol·licitat.

**Cas d'ús:** Comprar allotjament

**Actor Principal:** Client

**Àmbit:** Portal ISP

**Nivell d'objectiu:** Usuari

**Stakeholders i interessos:**

Usuari: Vol comprar un allotjament

**Precondició:** Ha d'estar identificat en el sistema.

**Garanties mínimes:** El sistema enregistra la petició de l'usuari.

**Garanties en cas d'èxit:** El sistema processa la petició i crea una comanda en el sistema.

**Escenari principal d'èxit:**

1. L'usuari indica que vol comprar un allotjament
2. El sistema li mostra un formulari per introduir el nom del domini
3. L'usuari indica que el nom de domini per al qual desitja comprar l'allotjament
4. El sistema enregistra la petició i crea la comanda

**Extensions:**

3a. L'usuari introdueix un domini no vàlid

3a1. El sistema li mostra el missatge "domini no vàlid" i torna a demanar el domini.



**Cas d'ús:** Modificar dades d'un client

**Actor Principal:** Treballador

**Àmbit:** Portal ISP

**Nivell d'objectiu:** Usuari

**Stakeholders i interessos:**

Usuari: Vol modificar les dades del sistema d'un client.

**Precondició:** Ha d'estar identificat en el sistema.

**Garanties mínimes:** El sistema enregistra la petició de l'usuari.

**Garanties en cas d'èxit:** El sistema processa la petició i modifica les dades el client.

**Escenari principal d'èxit:**

1. L'usuari indica que vol accedir a la base de clients
2. El sistema mostra el llistat de clients
3. L'usuari selecciona el client que vol modificar
4. El sistema mostra les dades del client en un formulari editable.
5. L'usuari introdueix les dades que vol modificar
6. El sistema valida les dades i les enregistra a la base de dades.

**Extensions:**

5a. L'usuari introdueix dades incoherents

5a1. El sistema mostra el missatge "Dades incoherents" i torna a mostrar el formulari

**Cas d'ús:** Activar Producte

**Actor Principal:** Treballador

**Àmbit:** Portal ISP

**Nivell d'objectiu:** Usuari

**Stakeholders i interessos:**

Usuari: Vol activar un producte d'un client

**Precondició:** Ha d'estar identificat en el sistema, el producte existeix , està assignat a un client i no està activat.

**Garanties mínimes:** El sistema enregistra la petició de l'usuari.

**Garanties en cas d'èxit:** El sistema processa la petició i activa el producte.

**Escenari principal d'èxit:**

1. L'usuari indica que vol veure la llista de clients
2. El sistema li mostra la llista
3. L'usuari selecciona el client per al qual vol gestionar productes
4. El sistema li mostra els productes que té contractats
5. L'usuari indica el producte que vol activar
6. El sistema activa el producte

**Extensions:**

3a. L'usuari selecciona un client sense productes

3a1. El sistema li indica que el client no té productes contractats.

**Cas d'ús:** Suspendre Producte

**Actor Principal:** Treballador

**Àmbit:** Portal ISP

**Nivell d'objectiu:** Usuari

**Stakeholders i interessos:**

Usuari: Vol suspendre un producte d'un client

**Precondició:** Ha d'estar identificat en el sistema, el producte existeix , està assignat a un client i està activat

**Garanties mínimes:** El sistema enregistra la petició de l'usuari.

**Garanties en cas d'èxit:** El sistema processa la petició i suspèn el producte.

**Escenari principal d'èxit:**

1. L'usuari indica que vol veure la llista de clients
2. El sistema li mostra la llista
3. L'usuari selecciona el client per al qual vol gestionar productes
4. El sistema li mostra els productes que té contractats
5. L'usuari indica el producte que vol suspendre
6. El sistema suspèn el producte

**Extensions:**

3a. L'usuari selecciona un client sense productes

3a1. El sistema li indica que el client no té productes contractats.

**Cas d'ús:** Aprovar Comanda

**Actor Principal:** Treballador

**Àmbit:** Portal ISP

**Nivell d'objectiu:** Usuari

**Stakeholders i interessos:**

Usuari: Vol aprovar una comanda

**Precondició:** Ha d'estar identificat en el sistema i existeix una comanda no aprovada

**Garanties mínimes:** El sistema enregistra la petició de l'usuari.

**Garanties en cas d'èxit:** El sistema processa la petició i aprova la comanda

**Escenari principal d'èxit:**

1. L'usuari indica que vol veure les comandes
2. El sistema mostra les comandes
3. L'usuari selecciona una comanda pendent
4. El sistema ofereix la opció d'aprovar o anul·lar la comanda
5. L'usuari indica que vol aprovar la comanda
6. El sistema aprova la comanda

**Extensions:**

2a No hi ha comandes pendents, el sistema mostra una llista buida

**Cas d'ús:** Anul·lar Comanda

**Actor Principal:** Treballador

**Àmbit:** Portal ISP

**Nivell d'objectiu:** Usuari

**Stakeholders i interessos:**

Usuari: Vol anul·lar una comanda

**Precondició:** Ha d'estar identificat en el sistema i existeix una comanda no aprovada

**Garanties mínimes:** El sistema enregistra la petició de l'usuari.

**Garanties en cas d'èxit:** El sistema processa la petició i anul·la la comanda

**Escenari principal d'èxit:**

1. L'usuari indica que vol veure les comandes
2. El sistema mostra les comandes
3. L'usuari selecciona una comanda pendent
4. El sistema ofereix la opció d'aprovar o anul·lar la comanda
5. L'usuari indica que vol anul·lar la comanda
6. El sistema anul·la la comanda

**Extensions:**

2a No hi ha comandes pendents, el sistema mostra una llista buida

**Cas d'ús:** Assignar Rol

**Actor Principal:** Administrador

**Àmbit:** Portal ISP

**Nivell d'objectiu:** Usuari

**Stakeholders i interessos:**

Usuari: Vol assignar un rol a un usuari del sistema

**Precondició:** Ha d'estar identificat en el sistema i existeix l'usuari pel qual es vol assignar el rol

**Garanties mínimes:** El sistema enregistra la petició de l'usuari.

**Garanties en cas d'èxit:** El sistema processa la petició i assigna el rol corresponent

**Escenari principal d'èxit:**

1. L'usuari indica que vol gestionar els usuari
2. El sistema mostra la llista d'usuaris del sistema
3. L'usuari selecciona l'usuari per al qual vol assignar un
4. El sistema mostra els rols disponibles
5. L'usuari indica quin rol vol assignar
6. El sistema assigna el rol a l'usuari.

**Extensions:**

**Cas d'ús:** Alta Producte

**Actor Principal:** Administrador

**Àmbit:** Portal ISP

**Nivell d'objectiu:** Usuari

**Stakeholders i interessos:**

Usuari: Vol donar d'alta un producte al catàleg

**Precondició:** Ha d'estar identificat en el sistema i tenir el rol d'administrador.

**Garanties mínimes:** El sistema enregistra la petició de l'usuari.

**Garanties en cas d'èxit:** El sistema processa la petició i publica un nou producte al catàleg

**Escenari principal d'èxit:**

1. L'usuari indica que vol gestionar el catàleg
2. El sistema mostra les opcions disponibles
3. L'usuari selecciona donar d'alta un producte
4. El sistema mostra un formulari amb les dades requerides per crear un nou producte ( Nom , Tipus: Domini o Hosting , Descripció, Preu, Plantilla Plesk , etc..)
5. L'usuari introdueix les dades requerides
6. El sistema valida les dades i dona d'alta el producte

**Extensions:**

- 5a. L'usuari introdueix dades incoherents al formulari
- 5a1. El sistema mostra de nou el formulari i indica l'error.

**Cas d'ús:** Modificar Producte

**Actor Principal:** Administrador

**Àmbit:** Portal ISP

**Nivell d'objectiu:** Usuari

**Stakeholders i interessos:**

Usuari: Vol modificar un producte al catàleg

**Precondició:** Ha d'estar identificat en el sistema i tenir el rol d'administrador.

**Garanties mínimes:** El sistema enregistra la petició de l'usuari.

**Garanties en cas d'èxit:** El sistema processa la petició i modifica el producte del catàleg

**Escenari principal d'èxit:**

1. L'usuari indica que vol gestionar el catàleg
2. El sistema mostra les opcions disponibles i la llista de productes
3. L'usuari selecciona el producte que vol modificar
4. El sistema mostra un formulari amb les dades existents de l producte ( Nom , Tipus: Domini o Allotjament , Descripció, Plantilla Plesk , etc..)
5. L'usuari introdueix les dades modificades
6. El sistema valida les dades i dona d'alta el producte

**Extensions:**

- 5a. L'usuari introdueix dades incoherents al formulari
- 5a1. El sistema mostra de nou el formulari i indica l'error.



**Cas d'ús:** Crear Servei

**Actor Principal:** Servidor d'Allotjament

**Àmbit:** Portal ISP

**Nivell d'objectiu:** Tasca

**Precondició:** Un treballador ha aprovat una comanda

**Garanties mínimes:** El servidor enregistra la petició d'aprovisionament

**Garanties en cas d'èxit:** El servidor enregistra la petició i crea el recurs assignat segons el tipus de producte que contingui la comanda

**Escenari principal d'èxit:**

1. El portal web envia una petició al servidor per crear un allotjament
2. El servidor d'allotjament crea el recurs sol·licitat
3. El portal web activa el producte relacionat del client.

**Extensions:**

- 1a . El portal web envia una petició errònia al servidor d'allotjament
- 1a1 El servidor d'allotjament retorna un codi d'error i rebutja la petició

**Cas d'ús:** Desactivar Servei

**Actor Principal:** Servidor d'Allotjament

**Àmbit:** Portal ISP

**Nivell d'objectiu:** Tasca

**Precondició:** Un treballador ha suspès un producte

**Garanties mínimes:** El sistema enregistra la petició cap al servidor d'allotjament

**Garanties en cas d'èxit:** El sistema enregistra la petició i el servidor desactiva el servei especificat

**Escenari principal d'èxit:**

1. El portal web envia una petició al servidor desactivar un servei
2. El servidor d'allotjament desactiva el servei especificat.
3. El portal web marca el producte del client com a suspès.

**Extensions:**

1a . El portal web envia una petició errònia al servidor d'allotjament

1a1 El servidor d'allotjament retorna un codi d'error i rebutja la petició

**Cas d'ús:** Activar Servei

**Actor Principal:** Servidor d'Allotjament

**Àmbit:** Portal ISP

**Nivell d'objectiu:** Tasca

**Precondició:** Un treballador ha activat un producte d'un client

**Garanties mínimes:** El sistema enregistra la petició cap al servidor d'allotjament

**Garanties en cas d'èxit:** El sistema enregistra la petició i el servidor activa el servei especificat

**Escenari principal d'èxit:**

4. El portal web envia una petició al servidor per activar un servei
5. El servidor d'allotjament activa el servei especificat.
6. El portal web marca el producte del client com a actiu.

**Extensions:**

- 1a . El portal web envia una petició errònia al servidor d'allotjament
- 1a1 El servidor d'allotjament retorna un codi d'error i rebutja la petició

## 5.4 ESQUEMA DE PANTALLES PRINCIPALS

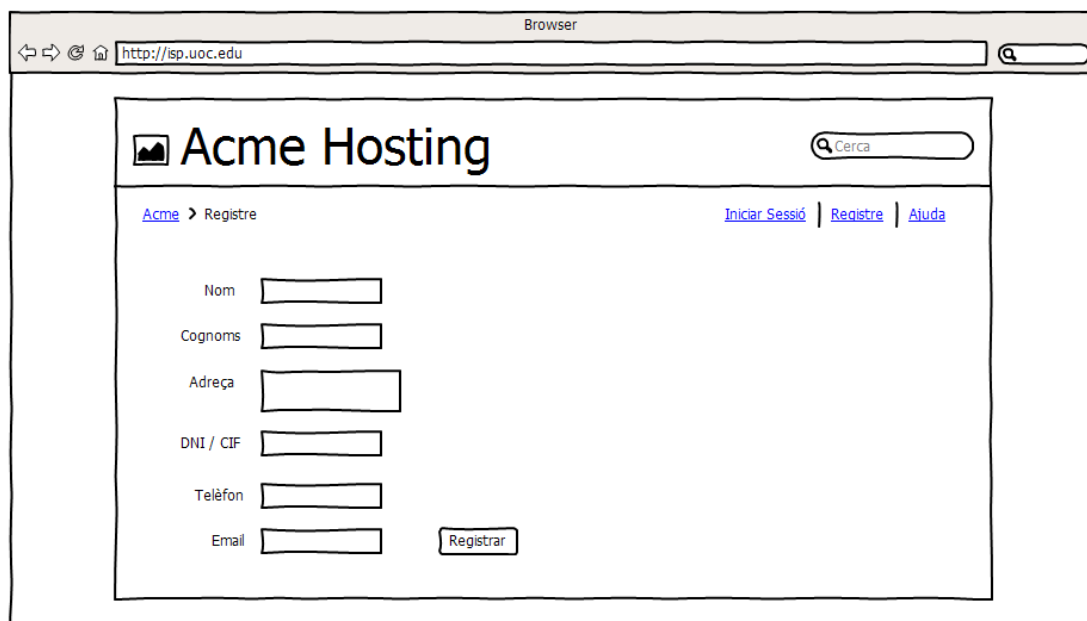
### 5.4.1 PORTADA

Es tracta de la pantalla inicial, la porta d'entrada de tots els usuaris al sistema.



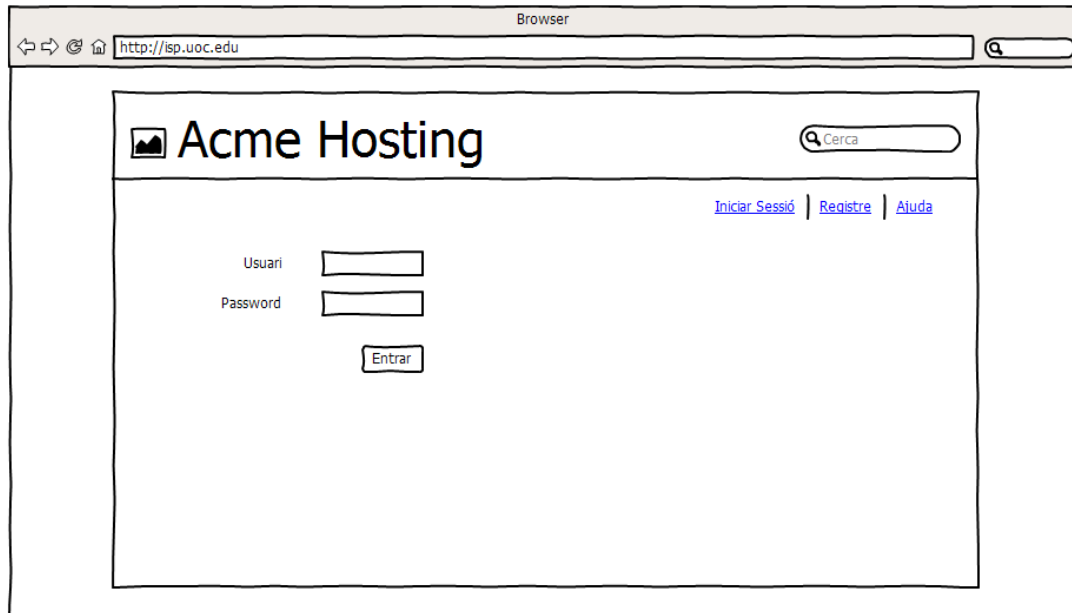
### 5.4.2 REGISTRE D'USUARI

És la pàgina on els usuaris poden crear una compta en el portal.

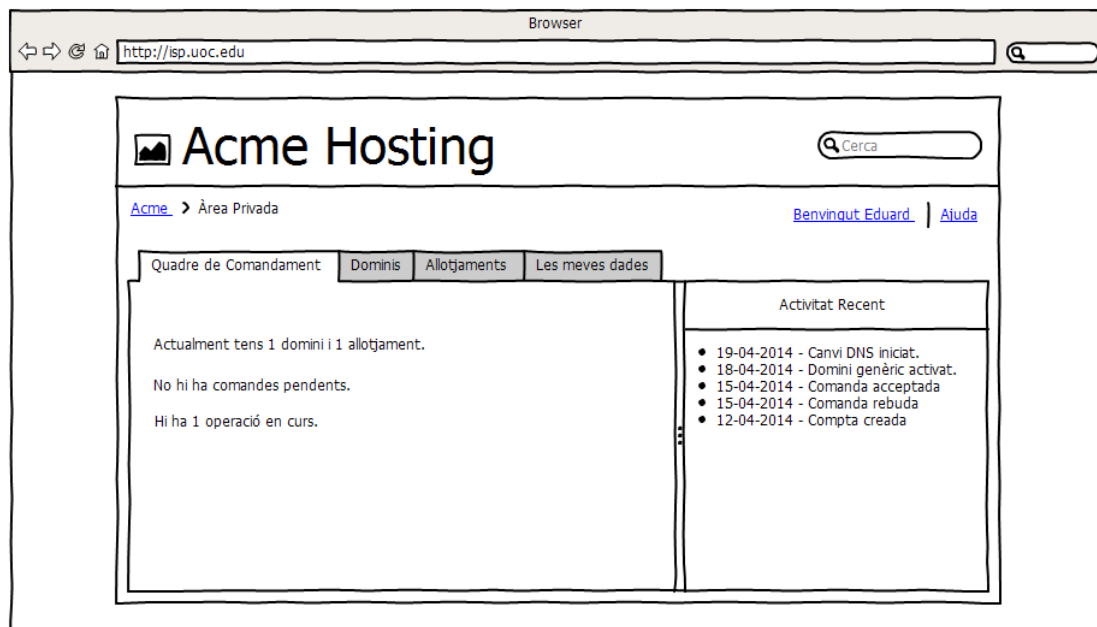


### 5.4.3 INICIAR SESSIÓ

En aquesta pantalla, els usuaris poden accedir a l'àrea privada mitjançant un nom d'usuari i una clau de pas.



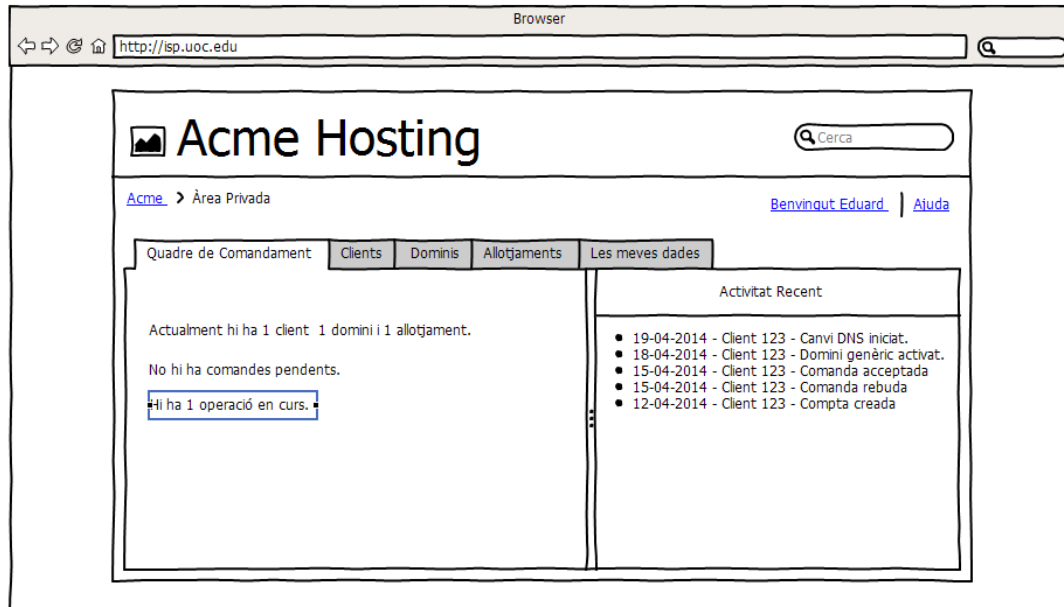
### 5.4.4 ÀREA PRIVADA – CLIENT



Els clients disposaran en primer lloc d'un quadre de comandament, on podran veure en primera instància, l'estat general de la seva compta. Això inclou la última activitat que ha tingut lloc en la seva compta i també, un resum general dels serveis que tenen contractats a l'empresa.

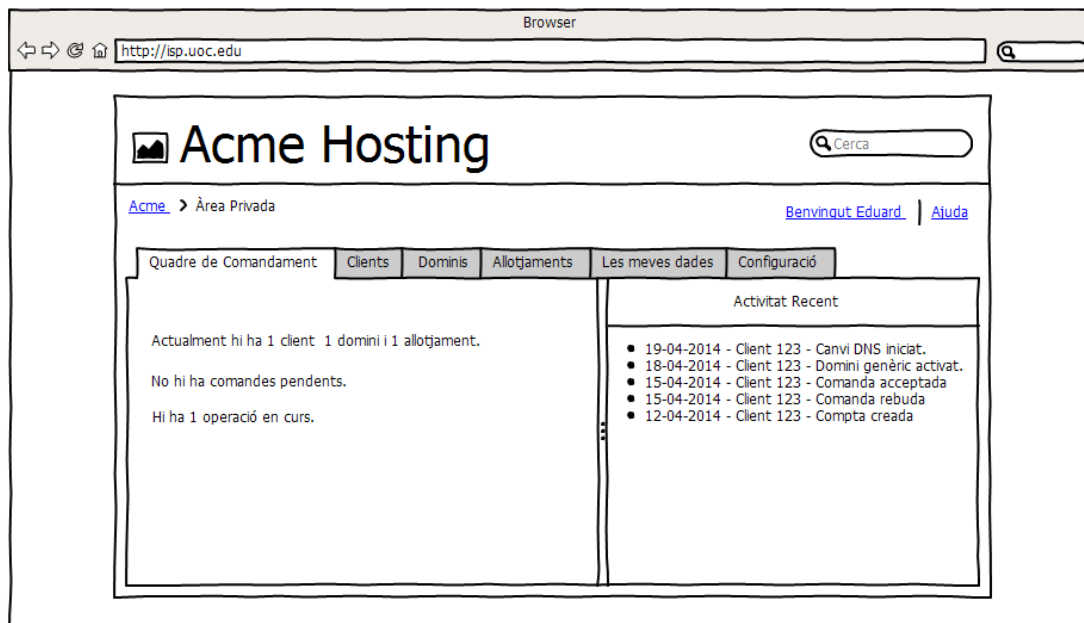
### 5.4.5 ÀREA PRIVADA TREBALLADOR

Molt similar a la pantalla del client , amb la diferència que els treballadors tenen accés a la base de dades de clients i no consulten els seus productes sinó els d'altres usuaris.



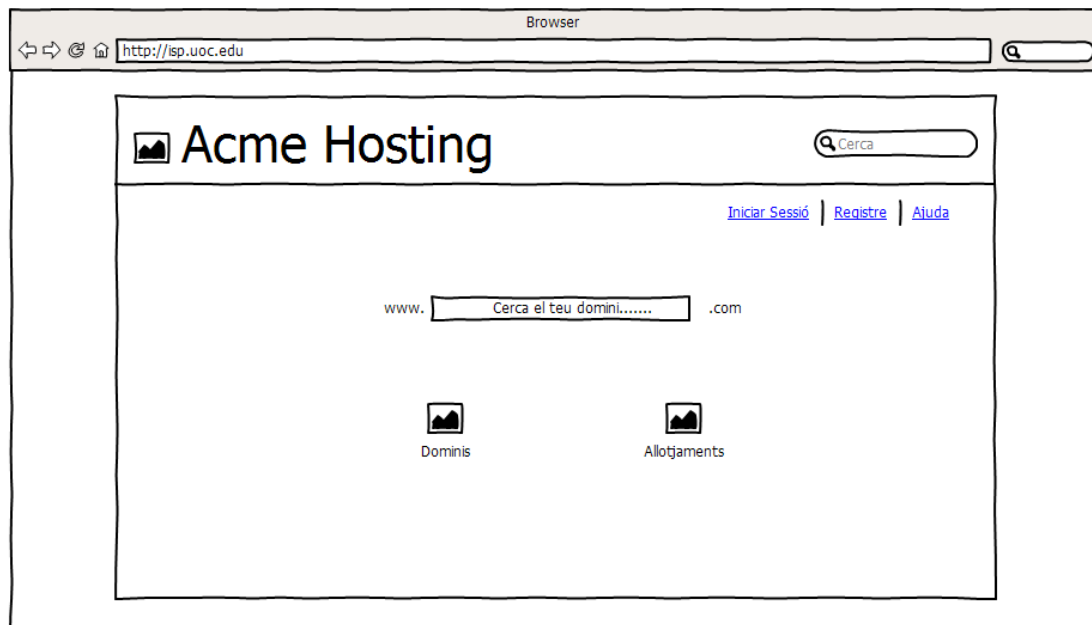
#### 5.4.6 ÀREA PRIVADA ADMINISTRADOR

L'àrea de l'administrador, tal i com passava amb la del treballador, té el mateix aspecte i distribució que la del client. En aquest cas, els administradors tenen accés complet a tot el sistema, totes les dades dels clients, a tots els productes. La diferència principal, és que els administradors tenen accés a la configuració del portal, lloc on es configuren les diferents connexions amb els servidors d'allotjament, o paràmetres com podrien ser el títol del portal i textos del sistema.

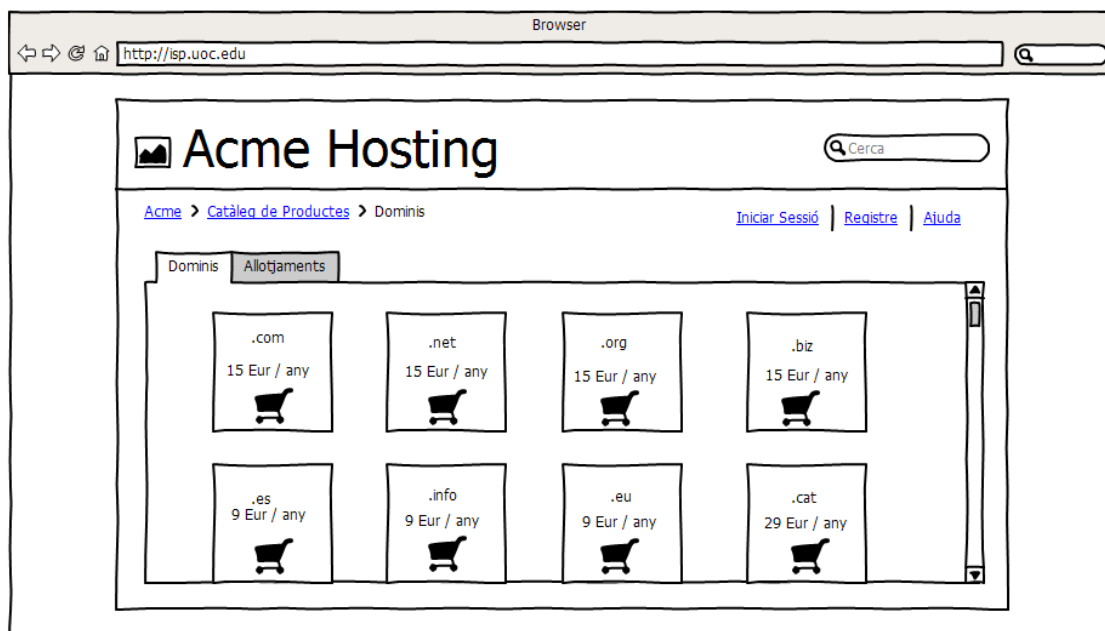


### 5.4.7 CATÀLEG DE PRODUCTES

El catàleg de productes es troba integrat en la portada, concretament, s'hi mostraran les diferents categories de productes que hi ha disponibles. En aquest projecte només s'utilitzaran dues categories: dominis i allotjaments.



### 5.4.8 CATÀLEG DE PRODUCTES - DOMINIS

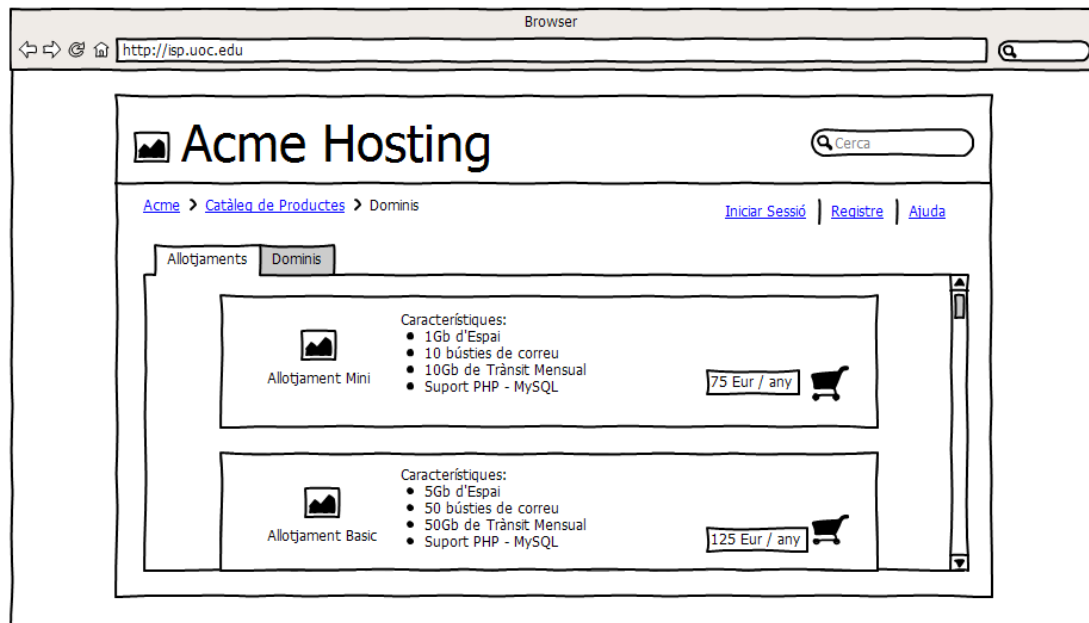


La vista de dominis , dins del catàleg, té un aspecte molt similar a una botiga en línia , cada producte correspon a una extensió de domini diferent.



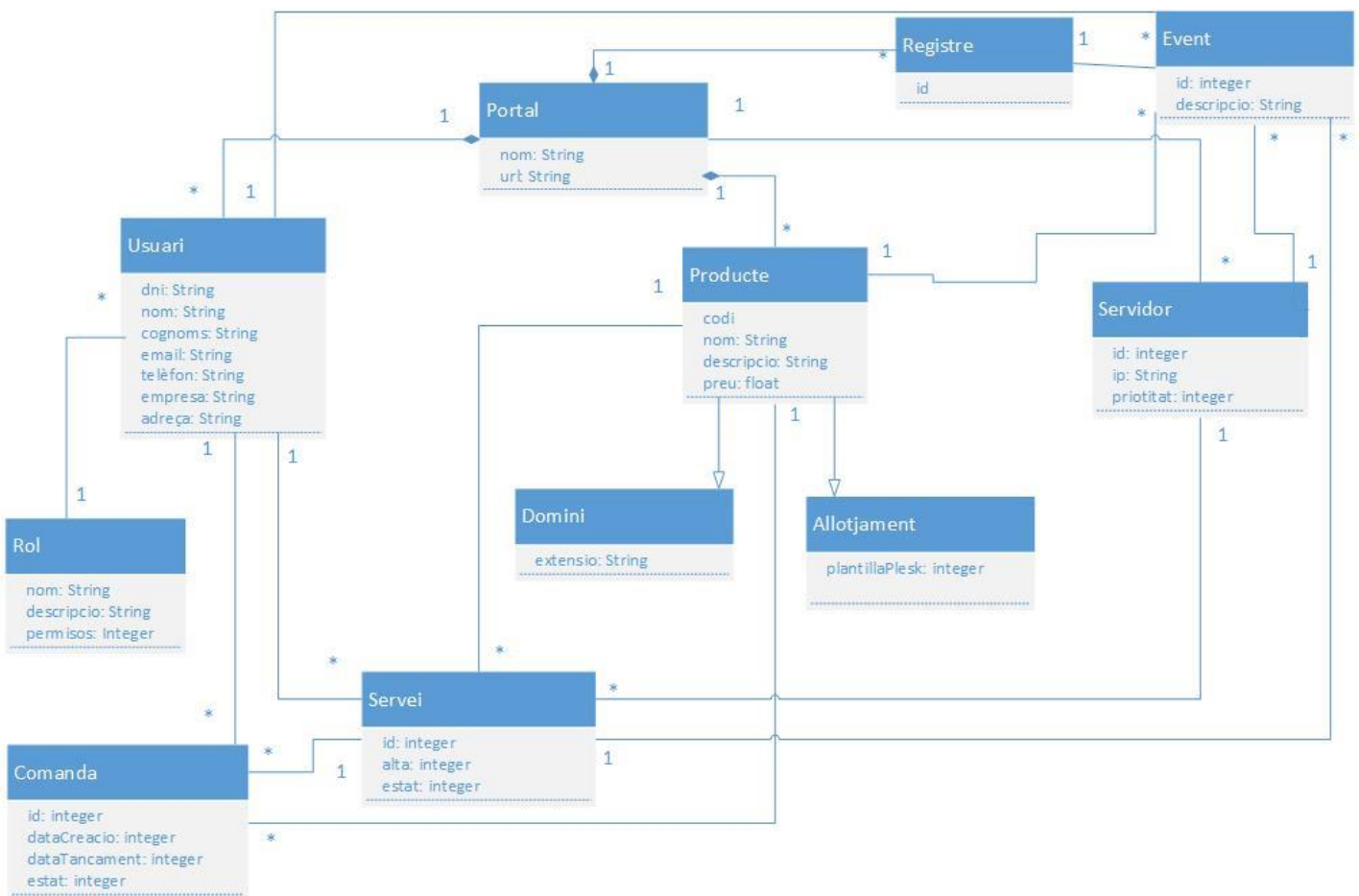
### 5.4.9 CATÀLEG DE PRODUCTES – ALLOTJAMENTS

La vista del catàleg d'allotjaments utilitza una distribució visual diferent a la de dominis degut al tipus de producte que s'ofereix. La diferència més remarcable és la descripció, que en el cas dels allotjaments és més extensa que no pas la dels dominis.



## 5.5 DIAGRAMA UML DE CLASSES

A continuació presentem el diagrama de les classes més destacades que conformen l'aplicació..

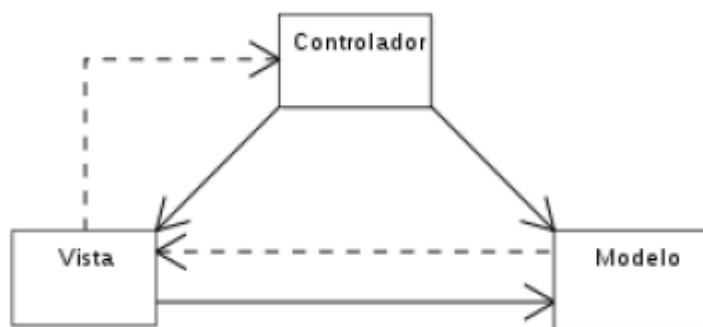


## 5.6 ARQUITECTURA

L'aplicació utilitza una arquitectura Model-Vista-Controlador(MVC) per tal de facilitar-ne el seu desenvolupament així com les seves futures millores.

Aquesta arquitectura es caracteritza per separar les dades i el seu tractament de la interfície de l'usuari, dividint l'aplicació en tres components principals:

- Model
- Vista
- Controlador



### 5.6.1 EL MODEL

El model és la representació de la informació amb la que treballa la nostra aplicació. Això inclou la gestió de tots als accessos a aquesta informació , així com les seves actualitzacions o també, la inserció de noves dades.

Aquest component defineix el que s'anomena la "lògica de negoci" , i la base de dades de MySQL serà un dels seus elements principals.

### 5.6.2 LA VISTA

La vista és la part que utilitzaran els usuaris per poder interactuar amb l'aplicació, en aquest component, hi trobarem tota la part relativa al disseny visual de l'aplicació.

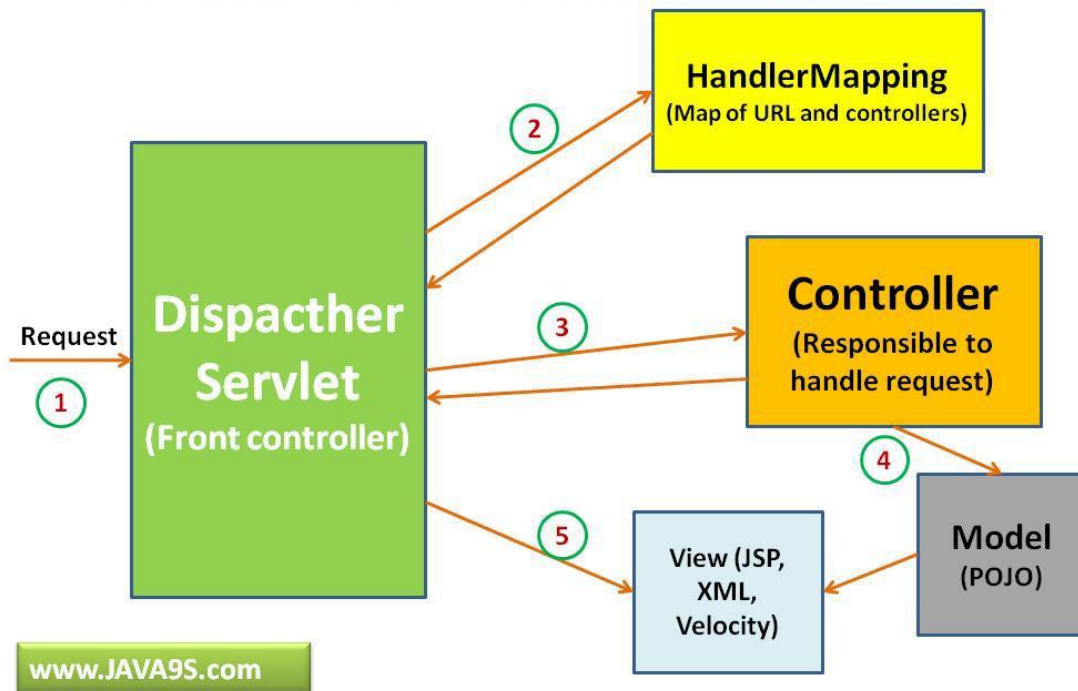
### 5.6.3 EL CONTROLADOR

El controlador és el component que accedeix al model per obtenir les dades i els fa arribar a la vista per tal que la informació sigui mostrada a l'usuari.

## 5.6.4 SPRING MVC

El framework Spring ens proporciona el seu propi model MVC que ofereix algunes particularitats respecte a l'arquitectura MVC tradicional.

## Spring 3 MVC- Basic Architecture



## 6. IMPLEMENTACIÓ I PRODUCTES OBTINGUTS

Tal com comentàvem a l'inici d'aquest document, no s'ha implementat la totalitat de l'abast del projecte, degut a la gran dimensió del mateix. Durant la fase d'implementació vaig haver de prendre la decisió de prioritzar la qualitat per davant de la quantitat, i la obtenció d'una base sòlida, extensible i que integrés, en la mesura del possible, tecnologies i estructures àmpliament utilitzades en aplicacions disponibles en el mercat avui en dia.

### 6.1 COMPONENTS DE L'APLICACIÓ

Passem a detallar els diferents components que formen el treball.

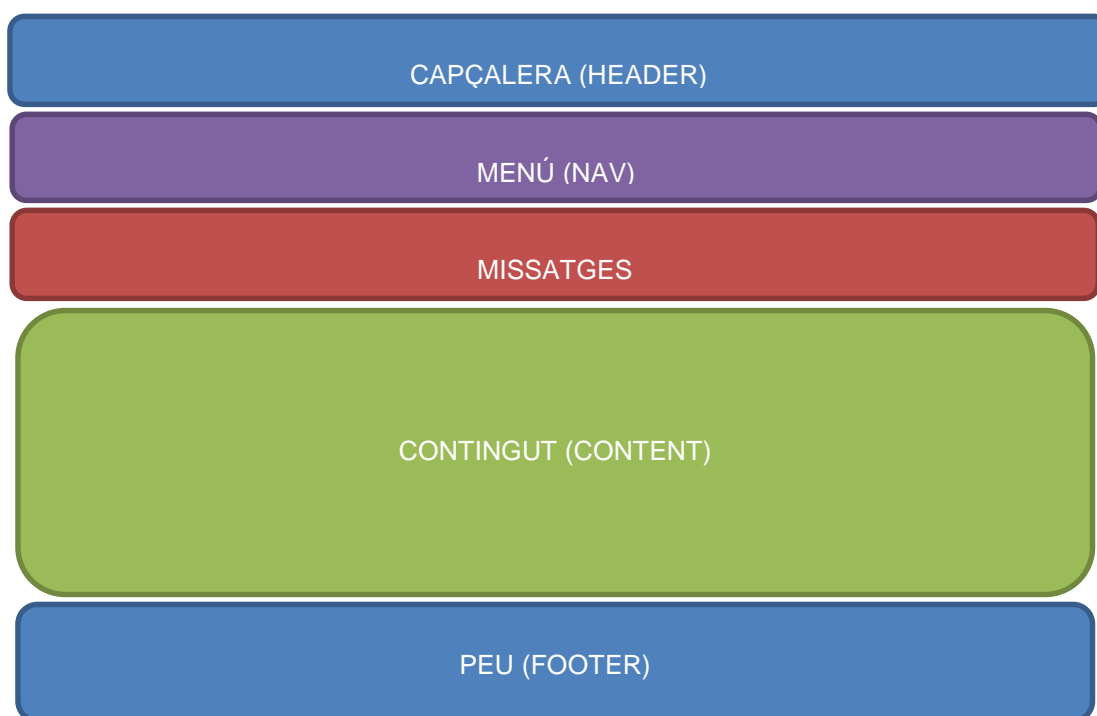
#### 6.1.1 LA VISTA – DISSENY DE LA INTERFÍCIE D'USUARI

##### 6.1.1.1 APACHE TILES

Per tal d'optimitzar el procés de desenvolupament, l'aplicació utilitza un sistema de plantilles mitjançant el framework "Apache Tiles". Aquest, es basa en el patró de disseny "Composite" i permet definir fragments de pàgines anomenats "tiles", els quals s'uneixen mitjançant les definicions de plantilla en un fitxer de configuració XML, formant la vista de pàgina final.

En aquesta aplicació s'ha implementat la següent plantilla de fragments:

##### 6.1.1.1.1 COMPOSICIÓ DE TILES



## 6.1.1.1.2 VISTA REAL DE PÀGINA

The screenshot shows a web portal for 'ACME Hosting'. At the top, the 'HEADER' contains the company name and a user greeting 'Hola client.' with buttons for 'Sortir' and 'Àrea Client'. Below this is the 'NAV' section with buttons for 'Dominis' and 'Allotjaments', and a shopping cart icon labeled '3 productes'. A green 'MISSATGES' (Messages) bar indicates 'Operació completada correctament!' with a 'Confirmar comanda' button. The 'CONTENT' area features a table of products:

#	Producte	Preu	
22	.cat	39,99	
28	Pro	150,0	
29	Revenedor	1000,0	

The 'FOOTER' at the bottom contains the text 'Treball Final de Carrera - UOC - ETIS - Eduard Gubert - Crèdits'.

## 6.1.1.1.3 ESTRUCTURA DE FITXERS

The file explorer shows the following structure:

- WebContent
  - META-INF
  - resources
  - WEB-INF
    - flows
    - js
    - layouts
      - default.xml 35 11/01/15 22:49
    - lib
    - portal-servlet.xml 32 11/01/15 22:49
    - templates
      - default.jsp 25 6/01/15 3:00
      - footer.jsp 20 4/01/15 13:20
      - header.jsp 26 6/01/15 3:50
      - missatges.jsp 25 6/01/15 3:50
      - nav.jsp 35 11/01/15 22:49
      - nav-admin.jsp 26 6/01/15 3:50
    - tiles
      - admin
        - allotjaments.jsp 35 11/01/15 22:49
        - cistella.jsp 35 11/01/15 22:49
        - credits.jsp 20 4/01/15 13:20
        - dominis.jsp 35 11/01/15 22:49
      - error
        - iniciwebflow.jsp 12 28/12/14 17:51
        - login.jsp 23 5/01/15 17:51
        - usuari.jsp 23 5/01/15 17:51
        - usuariCreat.jsp 9 27/12/14 17:51
  - web.xml 30 10/01/15 14:51

Tots els fitxers involucrats en el sistema de plantilles es troben dins la carpeta WebContent

#### Layouts – default.xml

Conté les definicions de plantilla per cada vista

#### Templates

Aquesta carpeta conté d'una banda, la distribució base de tiles, a l'arxiu default.jsp. De l'altra, trobem la programació de tots els fragments de la web que són comuns a la web pública o bé a l'àrea d'administració.

#### Tiles

Els fitxers d'aquesta carpeta corresponen als fragments de contingut, que es mostren al tile anomenat "content".

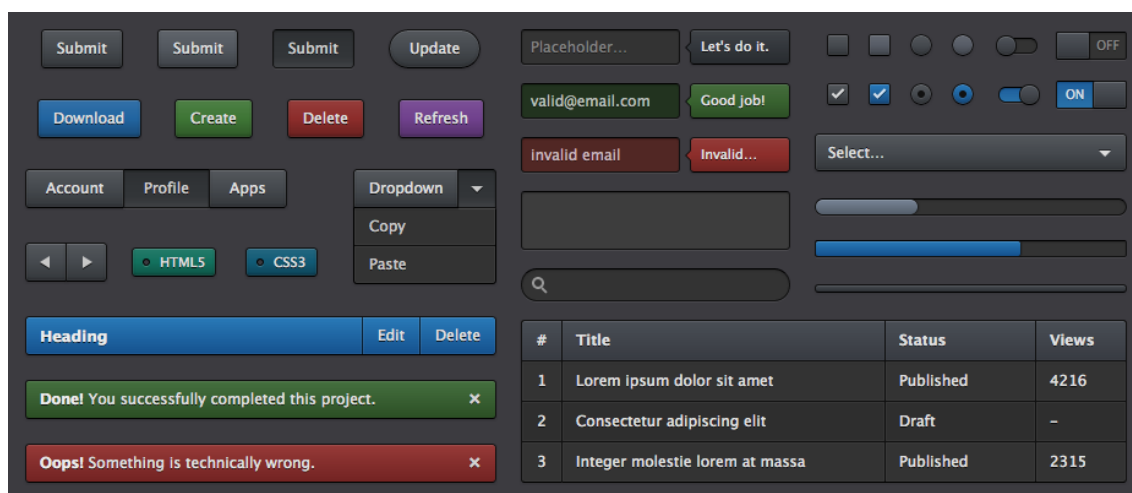
A l'arrel trobem els fragments de contingut de la web pública, i dins la carpeta admin, els que pertanyen a l'àrea d'administració o client.

La carpeta "error", conté els fragments utilitzats en el tractament d'excepcions.

### 6.1.1.2 CSS

La tecnologia CSS (Cascading Style Sheets) , és un llenguatge de fulls d'estil que s'utilitza per donar format als diferents elements que es visualitzen en l'aplicació.

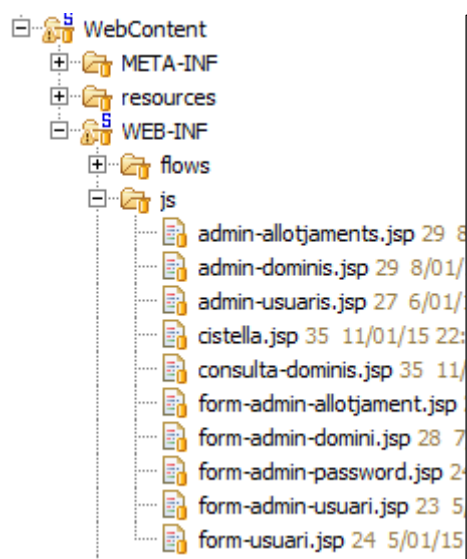
En aquest cas, part de les fulles d'estil que implementa el projecte estan basades en el següent kit CSS creat per Thibaut Couroble sota la llicència MIT.



### 6.1.1.3 JQUERY

La tecnologia de jQuery està basada en JavaScript, i ens permet de forma ràpida i amb poques línies de codi, interactuar amb els documents HTML, manipular events , desenvolupar animacions i afegir interacció mitjançant tècnics AJAX a les pàgines web.

#### 6.1.1.3.1 ESTRUCTURA DE FITXERS



Tots els fitxers que inclouen el codi jQuery, es troben a la carpeta js.

S'inclouen a la vista mitjançant el fitxer de definició de tiles default.xml , que hem vist anteriorment.

La seva nomenclatura, descriu la pàgina a on s'utilitzen de l'aplicació.

## 6.1.2 MÒDUL D'USUARIS

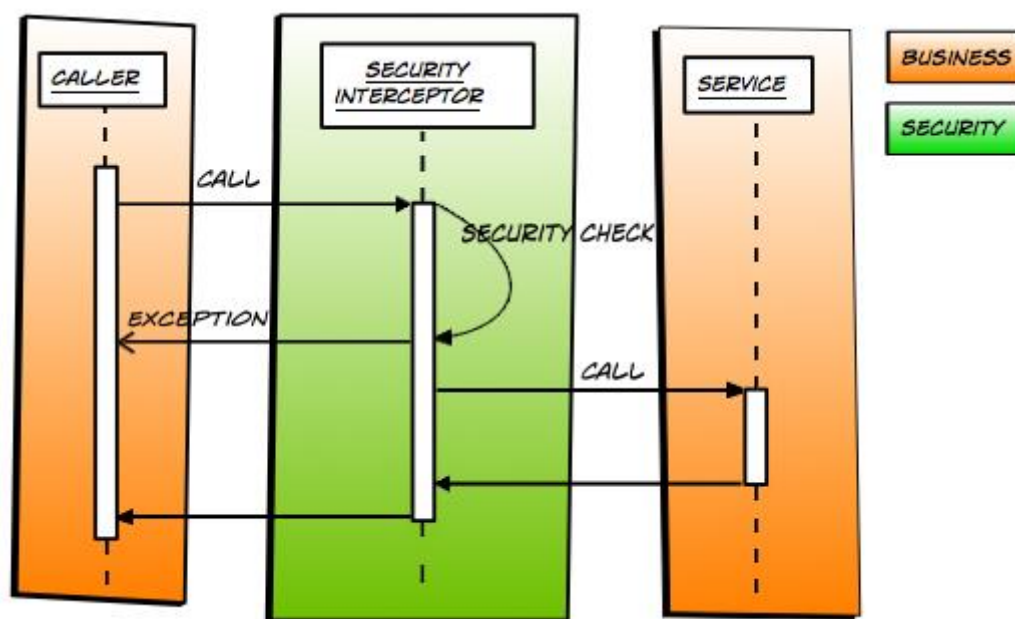
Aquest mòdul realitza totes les tasques relacionades amb la gestió d'usuaris:

- Creació de compta.
- Inici de sessió
- Rols d'usuari (Administrador i client)
- Crear, modificar i eliminar usuaris (només Administrador)

### 6.1.2.1 SPRING SECURITY

La implementació del mòdul d'usuaris s'ha realitzat mitjançant el framework Spring Security, integrant-lo a Spring MVC i Apache Tiles.

Aquest component "intercepta" totes les peticions que arriben a l'aplicació i accepta o denega l'accés al contingut en funció de la política de seguretat existent.



#### 6.1.2.1.1 ESTRUCTURA DE FITXERS



El fitxer de configuració d'spring security es troba dins el paquet \*.web.config i s'anomena security-context.xml.

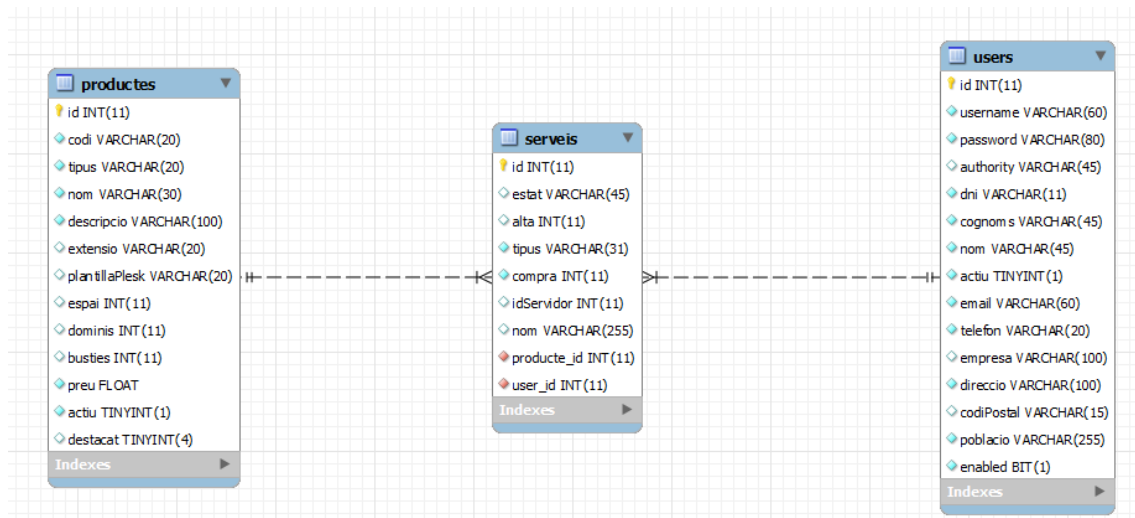


### 6.1.3 PERSISTÈNCIA DE DADES

La funcionalitat de persistència de dades s'ha implementat mitjançant Hibernate, un framework ORM (Object-Relational Mapping) i el sistema gestor de bases de dades MySQL.

#### 6.1.3.1 MODEL DE PERSISTÈNCIA

A continuació es mostra el disseny de la base de dades de l'aplicació.



#### 6.1.3.2 JERARQUIA DE CLASSES – ESTRATÈGIA DE TAULA ÚNICA

Hibernate ofereix diferents estratègies per tal d'implementar la persistència de jerarquia de classes:

- Taula única per jerarquia de classes
- Taula per subclasse
- Taula per classe

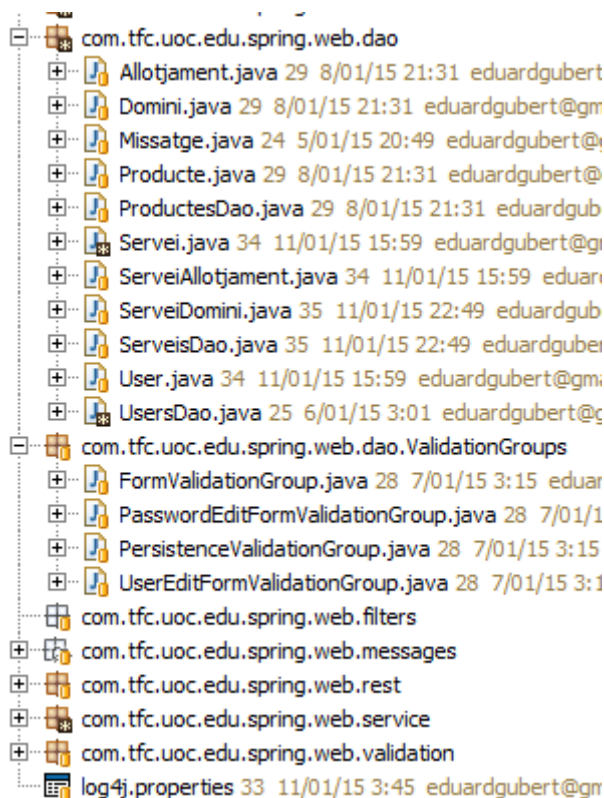
Pel desenvolupament del portal he optat per utilitzar la primera, que ofereix un millor rendiment a mesura que el volum de dades creix, ja que en aquest cas, no són necessàries comandes JOIN per accedir a les dades de les entitats.

La idea és simple, totes les entitats d'una jerarquia s'emmagatzemen en una mateixa taula. Per tal de diferenciar la classe nativa de cada d'objecte, s'utilitza un camp anomenat *discriminador*, que identifica el tipus d'entitat.

En el cas de la jerarquia de productes per exemple, el discriminador és el camp *tipus* de la taula *productes*, que pot tenir els valors "domini" o "allotjament".

Aquesta estratègia, té un punt feble, el fet d'utilitzar una mateixa taula per tota la jerarquia ens implica tenir camps amb valor nul a la base de dades, per tant, només podem realitzar-ne la validació en els formularis de l'aplicació.

## 6.1.3.3 ESTRUCTURA DE FITXERS



Tot el model de l'aplicació es troba dins el paquet \*.web.dao.

<Classe>.java

Per cada classe, s'ha implementat un fitxer per les operacions CRUD d'accés a la base de dades.. Aquests fitxers segueixen la nomenclatura:

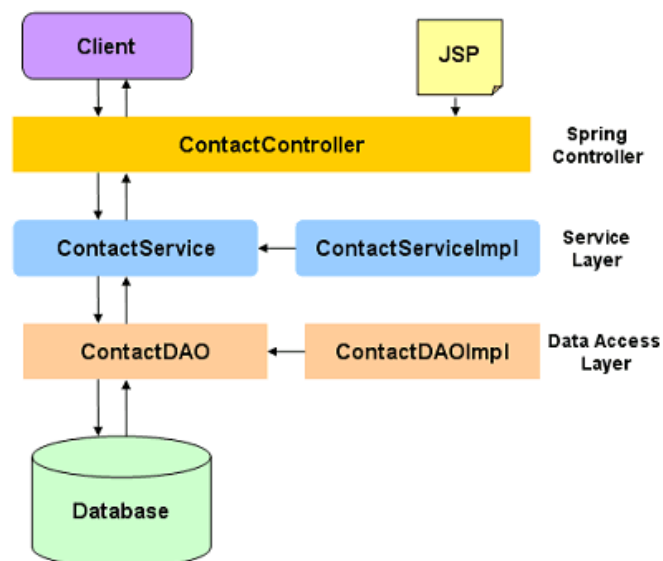
<Classe>Dao.java

Adicionalment, per poder utilitzar diferents criteris de validació de formularis, s'han creat diferents interfícies dins \*.web.dao.ValidationGroups.

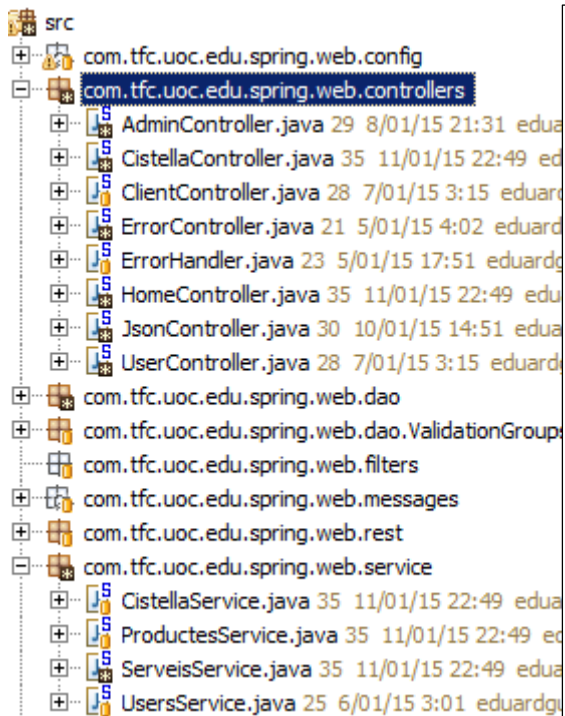
Cada formulari del portal té assignat un d'aquests grups, i només valida aquells camps que tenen el grup assignat.

## 6.1.4 CONTROLADORS I LA CAPA DE NEGOCI

El portal implementa múltiples controladors, dividits per àrees funcionals, per tal de facilitar la organització i comprensió del codi. Aquests es complementen amb les classes "Service" que són el component que utilitzen els controladors per tal d'accedir a les dades emmagatzemades a la base de dades.



### 6.1.4.1 ESTRUCTURA DE FITXERS



Els controladors es troben dins la carpeta \*.web.controllers, i els fitxers de la capa de servei dins la carpeta \*web.service

S'ha designat un control·lador per cadascuna de la següents funcionalitats:

- Panell Administració
- Àrea Client
- Cistella de la compra
- Pàgines d'error
- Home (Catàleg de productes)
- Webservice JSON
- Usuaris

### 6.1.5 CONSULTA DE DISPONIBILITAT DE DOMINIS

Aquest és un dels components clau de l'aplicació, permetent a l'usuari consultar la disponibilitat d'un domini en el que estigui interessat.

#### 6.1.5.1 TECNOLOGIA JSON – SPRING REST

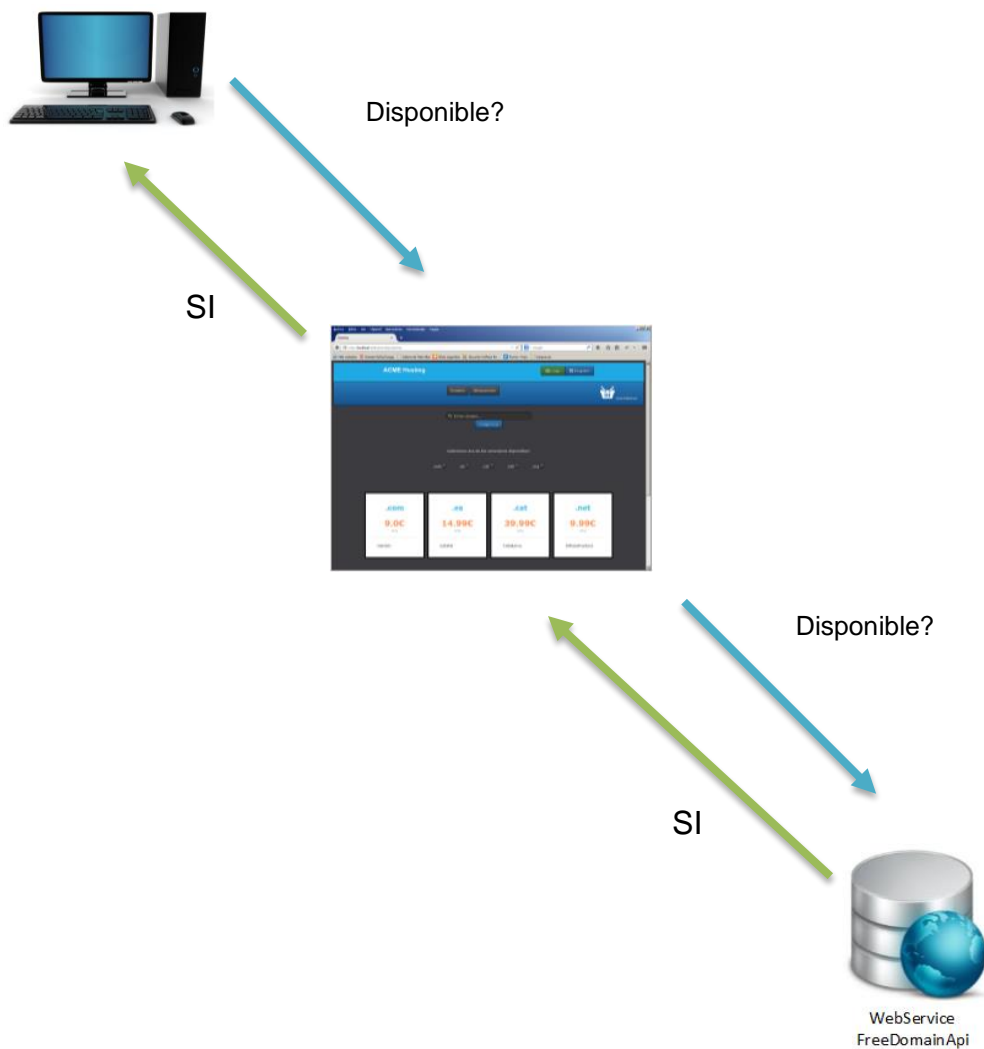
Per tal de poder oferir aquesta funcionalitat, s'ha implementat un client REST, que permet fer consultes de tipus JSON (Java Script Object Notation) a tercers. Més específicament, les consultes de disponibilitat de domini s'envien a un Webservice que ofereix l'empresa WhoAPI Inc. [www.freedomapi.com](http://www.freedomapi.com)

Spring ofereix un model específic que permet modelitzar les consultes JSON directament contra una entitat existent, en el nostre cas, la classe ConsultaDomini.

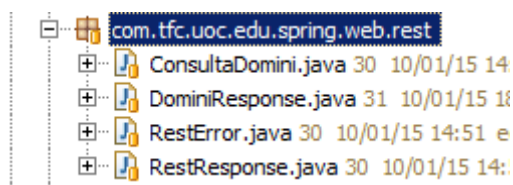
#### 6.1.5.2 WEBSERVICE – SERVIDOR JSON

Complementant el component anterior, el portal implementa la funcionalitat de servidor REST, que entrega les respostes que obté el client al navegador de l'usuari, per tal d'oferir el resultat de la consulta de disponibilitat.

6.1.5.3 DIAGRAMA DE COMUNICACIÓ



#### 6.1.5.4 ESTRUCTURA DE FITXERS



Tots els fitxers implicats en el client i servidor REST, es troben dins el paquet \*.web.rest.

Es necessita també el controlador JsonController que hem vist en l'apartat anterior.

#### 6.1.6 TESTING I LOGGING

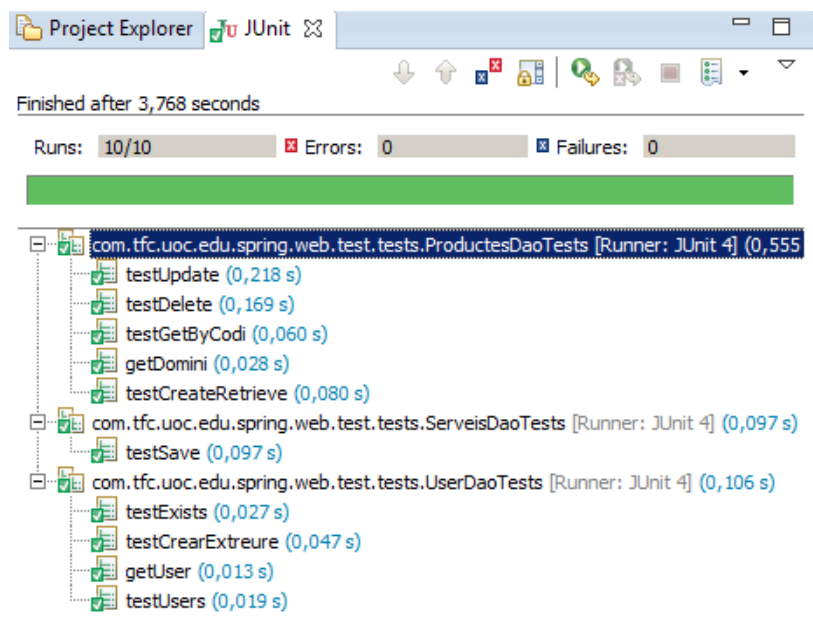
Amb l'objectiu de reduir els errors i facilitar la depuració de l'aplicació s'han integrat 2 components àmpliament utilitzats en el món J2EE. Es tracta de JUnit i Log4j.

##### 6.1.6.1 JUNIT

JUnit és un framework que permet la programació de unitats de test que, en el context d'aquesta aplicació, serveixen per validar el correcte funcionament de la capa de persistència de dades.

De fet, el component JUnit funciona de manera independent a l'execució del portal, utilitzant una base de dades separada anomenada PORTAL\_TEST, per realitzar les proves. No només això, utilitza també el seu propi context amb un perfil anomenat "dev", en contraposició al perfil de context "production" que utilitzen la resta de components.

Exemple d'execució dels jocs de proves:



### 6.1.6.1.1 ESTRUCTURA DE FITXERS

JUnit es troba en un paquet apart de la resta de l'aplicació , i es compon dels següents sub-paquets:

- \***.web.test.config**: Fitxers de configuració
- \***.web.test.tests**: Jocs de proves

### 6.1.6.2 LOG4J

La integració de les llibreries de Log4j permet agrupar i filtrar els missatges, també anomenats “logs”, que genera l'aplicació en temps d'execució.

#### 6.1.6.2.1 ESTRUCTURA DE FITXERS

Log4j disposa d'un únic fitxer de configuració , situat a l'arrel del projecte, anomenat *log4j.properties*.

### 6.1.7 LOCALITZACIÓ DE MISSATGES

Finalment, l'últim component a destacar , és la implementació bàsica d'un sistema de localització de missatges, és a dir. L'aplicació disposa de l'estructura necessària per tal de centralitzar tots els missatges del sistema en un únic fitxer i, d'aquesta manera, facilitar-ne la seva traducció a altres idiomes en cas que fos útil en un futur.

#### 6.1.7.1 ESTRUCTURA DE FITXERS

Tots els missatges es troben dins el paquet \*.web.messages, en el fitxer *messages.properties*

## 7. MODIFICACIONS DEL PLANTEJAMENT INICIAL

### 7.1 MODEL UML

Una de les decisions claus que he hagut de prendre durant el transcurs del projecte, ha estat la d'escollir una estratègia d'implementació de la persistència de dades per les classes amb especialitzacions que en aquest projecte són: *Productes* i *Serveis*.

Inicialment, durant l'etapa d'anàlisi es va preveure ja realitzar una especialització per la classe *Productes* que s'especialitzava en dues sub-classes: *Dominis*, i *Allotjaments*. Aquesta, de fet, és una de les complexitats més remarcables del projecte ja que, a diferència d'una botiga on-line tradicional, els productes són conceptualment diferents entre si i, en el cas dels dominis, es requereixen diferents comprovacions abans de poder "servir" el producte.

- L'article es personalitza amb un nom
- No serveix qualsevol nom, en funció de l'extensió del domini, pot estar lliure o no per comprar-lo.

Seguint aquesta problemàtica, vaig decidir realitzar un canvi en el model de classes, que afecta a les entitats *Servei* i *Comanda*, eliminant aquesta última, i utilitzant *Servei*, com a superclasse de dues noves entitats *ServeiDomini* i *ServeiAllotjament*.

Amb aquesta estructura, és possible agrupar d'una forma àgil i fàcilment extensible les funcionalitats del portal, mantenint sempre la coherència entre entitats, que queden agrupades sota la seva finalitat, ja sigui un servei contractat per un client o un producte que s'ofereix al catàleg del portal.

### 7.2 ABAST DEL PROJECTE

En la part final del projecte m'he vist obligat a reduir-ne el seu abast, donada l'extensió del plantejament inicial, prioritant l'entrega de casos d'ús funcionals i acabats, així com la construcció d'una base sòlida i viable de l'aplicació en un possible entorn de producció.

## 8. CONCLUSIONS

Valoro molt positivament la realització d'aquest projecte donat que m'ha permès adquirir coneixements molt valuosos sobre la tecnologia J2EE, així com els diferents recursos que ofereix la tecnologia Spring o altres components que he esmentat en aquest document.

Un dels punts forts d'aquesta plataforma de desenvolupament és, sense cap mena de dubtes, la quantitat de llibreries de codi obert disponibles, així com la estructura resultant del projecte: ordenada, fàcilment extensible i de fàcil comprensió

Com a punt negatiu, destacaria que la corba d'aprenentatge és molt accentuada en la fase inicial, de fet, haig de remarcar que el temps invertit en formació i documentació ha estat molt superior al temps dedicat a la implementació d'aquest projecte.

## 9. GLOSSARI

**Hosting:** ordinador o conjunt d'ordinadors que ofereixen serveis, dades...a la resta d'ordinadors connectat a la xarxa, ja sigui local o global.

**ISP:** són les sigles de *Internet Service Provider*. És un proveïdor que ofereix principalment servei d'accés a Internet, afegint-hi altres serveis relacionats, com ara correu electrònic, allotjament web o blogs

**J2EE:** són les sigles de Java 2 Enterprise Edition que és l'edició empresarial del paquet Java creat i distribuït per Sun Microsystems. Comprenen un conjunt d'especificacions i funcionalitats orientades al desenvolupament d'aplicacions empresarials. A causa de que J2EE no deixa de ser un estàndard, existeixen altres productes creats a partir d'ella encara que no exclusivament.

**Arquitectura Model–View–Controller (MVC):** és un patró de disseny utilitzat per a la implementació d'interfícies d'usuari. Aquest patró de desenvolupament de programari divideix l'aplicació en tres parts interconnectades: el model de dades, la interfície usuari i la lògica de control. El patró MVC es veu freqüentment en aplicacions web, on es pot visualitzar una pàgina HTML i el codi que proveeix de dades dinàmiques a la pàgina, el controlador és el sistema de gestió de bases de dades i el model és el model de dades.

**Eclipse:** programa per a desenvolupar aplicacions amb Java (i altres).

**Spring:** És l'abreviatura de Spring framework, que és un marc de treball de codi obert per la plataforma Java.



**Hibernate:** és una solució implementada pel mapeig d'objectes relacionals (ORM) per aplicacions Java, sobre una base de dades relacional. Els seus propòsits bàsics són els d'alliberar el programador d'un seguit de tasques pròpies de la persistència de dades relacionals i dotar les aplicacions de portabilitat entre SGBDs diferents.

**Apache Tomcat:** funciona com un contenidor de servlets creats sota el projecte Jakarta a l'Apache Software Foundation. Tomcat implementa les especificacions dels servlets i de JavaServer Pages (JSP) de Sun Microsystems. Es considera un servidor d'aplicacions.

**Mysql:** sistema de gestió de base de dades, habitualment utilitzada en entorn Linux i amb programació PHP. Permet, per exemple, des de la creació de qualsevol sistema de registre d'usuaris, fins les funcionalitats més avançades d'un projecte web.

**Allotjament:** servei web que permet a qualsevol internauta pujar fàcilment qualsevol tipus de fitxer des del seu disc dur al servidor d'allotjament.

**Domini:** nom assignat a una o més adreces IP

## 10. BIBLIOGRAFIA

[http://es.wikipedia.org/wiki/Java\\_EE](http://es.wikipedia.org/wiki/Java_EE)

<http://spring.io/>

<http://tomcat.apache.org/>

<http://download1.parallels.com/Plesk/Doc/en-US/online/plesk-api-rpc/index.htm?fileName=33852.htm>

<http://hibernate.org/>

[www.udemy.com](http://www.udemy.com)

<http://sammy-woo.blogspot.com.es/2011/07/t3.html>

<http://viralpatel.net/blogs/hibernate-inheritance-table-per-hierarchy-mapping/>

<http://viralpatel.net/blogs/spring3-mvc-hibernate-maven-tutorial-eclipse-example/>

## 11. ANNEX 1 – INSTAL·LACIÓ DE L'APLICACIÓ

### 1.Requeriments

-----

Apache Tomcat 7  
Servidor MySQL 5.X

### 2.Instal·lació DB

-----

Importar fitxer install.sql

### 3.Configuració JDBC Producció

-----

Configurar connexió jdbc dins el projecte fitxer  
/portalisp/src/com/tfc/uoc/edu/spring/web/config/jdbc.properties

```
jdbc.username = portal  
jdbc.password = 1234567890  
jdbc.driver = com.mysql.jdbc.Driver  
jdbc.url = jdbc:mysql://localhost:3306/portal
```

### 4. Configuració JDBC Test JUnit (OPCIONAL)

-----

/portalisp/test/com/tfc/uoc/edu/spring/web/test/config/jdbc.properties

```
jdbc.username = portal  
jdbc.password = 1234567890  
jdbc.driver = com.mysql.jdbc.Driver  
jdbc.url = jdbc:mysql://localhost:3306/portal_test
```

### 5. AJUSTAMENTS TOMCAT (OPCIONAL)

-----

Per a un funcionament òptim del sistema es recomana indicar els següents paràmetres a la instància del servidor Tomcat:

-XX:PermSize=256m -XX:MaxPermSize=512m

### 6. USUARIS DE PROVA

-----

admin - 12345 (Administrador)  
client - 12345 (Client)

-----  
Repositori públic

-----  
<http://code.google.com/p/portal-isp/>

## 12. ANNEX 2 – SCRIPT CREACIÓ DB I DADES DE MOSTRA

```

CREATE DATABASE IF NOT EXISTS `portal_test` /*!40100 DEFAULT CHARACTER SET
latin1 COLLATE latin1_spanish_ci */;
USE `portal_test`;
-- MySQL dump 10.13 Distrib 5.6.13, for Win32 (x86)
--
-- Host: localhost Database: portal
-----
-- Server version 5.6.17-log

/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
/*!40101 SET NAMES utf8 */;
/*!40103 SET @OLD_TIME_ZONE=@@TIME_ZONE */;
/*!40103 SET TIME_ZONE='+00:00' */;
/*!40014 SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0 */;
/*!40014 SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS,
FOREIGN_KEY_CHECKS=0 */;
/*!40101 SET @OLD_SQL_MODE=@@SQL_MODE,
SQL_MODE='NO_AUTO_VALUE_ON_ZERO' */;
/*!40111 SET @OLD_SQL_NOTES=@@SQL_NOTES, SQL_NOTES=0 */;

--
-- Table structure for table `config`
--

DROP TABLE IF EXISTS `config`;
/*!40101 SET @saved_cs_client = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `config` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `parametre` varchar(45) CHARACTER SET utf8 NOT NULL,
  `valor` varchar(200) CHARACTER SET utf8 NOT NULL,
  `descripcio` varchar(200) CHARACTER SET utf8 NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 COLLATE=latin1_spanish_ci;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Dumping data for table `config`
--

LOCK TABLES `config` WRITE;
/*!40000 ALTER TABLE `config` DISABLE KEYS */;
/*!40000 ALTER TABLE `config` ENABLE KEYS */;
UNLOCK TABLES;

--
-- Table structure for table `productes`
--

DROP TABLE IF EXISTS `productes`;
/*!40101 SET @saved_cs_client = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `productes` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `codi` varchar(20) CHARACTER SET utf8 NOT NULL,

```

```

`tipus` varchar(20) CHARACTER SET utf8 NOT NULL,
`nom` varchar(30) CHARACTER SET utf8 NOT NULL,
`descripcio` varchar(100) CHARACTER SET utf8 NOT NULL,
`extensio` varchar(20) CHARACTER SET utf8 DEFAULT NULL,
`plantillaPlesk` varchar(20) CHARACTER SET utf8 DEFAULT NULL,
`espai` int(11) DEFAULT NULL,
`dominis` int(11) DEFAULT NULL,
`busties` int(11) DEFAULT NULL,
`preu` float NOT NULL,
`actiu` tinyint(1) NOT NULL DEFAULT '1',
`destacat` tinyint(4) DEFAULT NULL,
PRIMARY KEY (`id`),
UNIQUE KEY `codi_UNIQUE` (`codi`)
) ENGINE=InnoDB AUTO_INCREMENT=67 DEFAULT CHARSET=latin1
COLLATE=latin1_spanish_ci;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Dumping data for table `productes`
--

LOCK TABLES `productes` WRITE;
/*!40000 ALTER TABLE `productes` DISABLE KEYS */;
INSERT INTO `productes` VALUES
(55,'com','Domini','.com','Genèric','com',NULL,NULL,NULL,NULL,9,1,NULL),(56,'personal-
linux','Allotjament','Personal','Linux - PHP - MySQL ',NULL,NULL,1,1,1,80,1,0),(59,'linux-
pro','Allotjament','Pro','Linux - PHP - MySQL',NULL,NULL,5,5,20,150,1,1),(62,'linux-
revededor','Allotjament','Revededor','Linux - PHP -
MySQL',NULL,NULL,100,100,1000,1000,1,0),(63,'es','Domini','.es','Estatl','es',NULL,NULL,NU
LL,NULL,14.99,1,NULL),(64,'cat','Domini','.cat','Catalunya','cat',NULL,NULL,NULL,NULL,39.99,
1,NULL),(65,'net','Domini','.net','Infraestructura','net',NULL,NULL,NULL,NULL,9.99,1,NULL),(66,'
org','Domini','.org','Organitzacions','org',NULL,NULL,NULL,NULL,12.99,1,NULL);
/*!40000 ALTER TABLE `productes` ENABLE KEYS */;
UNLOCK TABLES;

--
-- Table structure for table `serveis`
--

DROP TABLE IF EXISTS `serveis`;
/*!40101 SET @saved_cs_client = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `serveis` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `estat` varchar(45) CHARACTER SET utf8 DEFAULT NULL,
  `alta` int(11) DEFAULT NULL,
  `tipus` varchar(31) COLLATE latin1_spanish_ci NOT NULL,
  `compra` int(11) NOT NULL,
  `idServidor` int(11) DEFAULT NULL,
  `nom` varchar(255) COLLATE latin1_spanish_ci DEFAULT NULL,
  `producte_id` int(11) NOT NULL,
  `user_id` int(11) NOT NULL,
  PRIMARY KEY (`id`),
  KEY `FK7643B87935C925CA` (`producte_id`),
  KEY `FK7643B879E7867DAA` (`user_id`),
  CONSTRAINT `FK7643B879E7867DAA` FOREIGN KEY (`user_id`) REFERENCES `users`
(`id`),
  CONSTRAINT `FK7643B87935C925CA` FOREIGN KEY (`producte_id`) REFERENCES
`productes` (`id`)

```

```

) ENGINE=InnoDB AUTO_INCREMENT=30 DEFAULT CHARSET=latin1
COLLATE=latin1_spanish_ci;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Dumping data for table `serveis`
--

LOCK TABLES `serveis` WRITE;
/*!40000 ALTER TABLE `serveis` DISABLE KEYS */;
INSERT INTO `serveis` VALUES (22,'CISTELLA',0,'Domini',0,NULL,'terra-
golfa',64,3),(28,'CISTELLA',0,'Allotjament',0,0,NULL,59,3),(29,'CISTELLA',0,'Allotjament',0,0,NU
LL,62,3);
/*!40000 ALTER TABLE `serveis` ENABLE KEYS */;
UNLOCK TABLES;

--
-- Table structure for table `users`
--

DROP TABLE IF EXISTS `users`;
/*!40101 SET @saved_cs_client = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `users` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `username` varchar(60) CHARACTER SET utf8 NOT NULL,
  `password` varchar(80) CHARACTER SET utf8 NOT NULL,
  `authority` varchar(45) CHARACTER SET utf8 DEFAULT 'ROLE_USER',
  `dni` varchar(11) CHARACTER SET utf8 NOT NULL,
  `cognoms` varchar(45) CHARACTER SET utf8 NOT NULL,
  `nom` varchar(45) CHARACTER SET utf8 NOT NULL,
  `actiu` tinyint(1) NOT NULL DEFAULT '1',
  `email` varchar(60) CHARACTER SET utf8 NOT NULL,
  `telefon` varchar(20) CHARACTER SET utf8 NOT NULL,
  `empresa` varchar(100) CHARACTER SET utf8 DEFAULT NULL,
  `direccio` varchar(100) CHARACTER SET utf8 NOT NULL,
  `codiPostal` varchar(15) CHARACTER SET utf8 DEFAULT NULL,
  `poblacio` varchar(255) CHARACTER SET utf8 NOT NULL,
  `enabled` bit(1) NOT NULL,
  PRIMARY KEY (`id`),
  UNIQUE KEY `username_UNIQUE` (`username`),
  UNIQUE KEY `dni_UNIQUE` (`dni`)
) ENGINE=InnoDB AUTO_INCREMENT=4 DEFAULT CHARSET=latin1
COLLATE=latin1_spanish_ci;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Dumping data for table `users`
--

LOCK TABLES `users` WRITE;
/*!40000 ALTER TABLE `users` DISABLE KEYS */;
INSERT INTO `users` VALUES
(2,'admin','2744075e7e422534a58389f8b501aa2dbe3f62d6066d8b38e894d73a47ff42debde92
402710fbbebb','ROLE_ADMIN','40536946K','Gubert
Bril-li','Eduard',1,'eduardgubert@hotmail.com','657851100','Eduard Gubert','C/Bòbila 3, 1er
2a','17242','Quart',' '), (3,'client','2744075e7e422534a58389f8b501aa2dbe3f62d6066d8b38e89
4d73a47ff42debde92402710fbbebb','ROLE_USER','123456789K','Ramon
Gràcia','Jordi',1,'jramon@tfc.uoc','93 390 390 87','Turrans Ramon S.A','Av Barcelona, 110 1er
2a','080880','Barcelona',' ');

```

```
/*!40000 ALTER TABLE `users` ENABLE KEYS */;  
UNLOCK TABLES;  
/*!40103 SET TIME_ZONE=@OLD_TIME_ZONE */;  
  
/*!40101 SET SQL_MODE=@OLD_SQL_MODE */;  
/*!40014 SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS */;  
/*!40014 SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS */;  
/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;  
/*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;  
/*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;  
/*!40111 SET SQL_NOTES=@OLD_SQL_NOTES */;
```