

Enomarket, botiga virtual de vins mitjançant tecnologia J2EE

Raül Melià Gracia

ETIS

Vicenç Font Sagristà

20 de Juny de 2011

Dedicatòria i agraïments

Són moltes les estones d'estudi al llarg de tants anys i moltes les hores que no he pogut dedicar completament a la família, i tinc molt present que sense ells això no hagués estat possible, en els moments difícils la comprensió i el recolzament han estat màxims.

En primer lloc, agrair als consultors i companys d'aula, l'ajuda que m'han donat al llarg de tants semestres.

En segon lloc, pel seu suport, a tota la meva família: pares, sogres, cosins, cunyats i amics, per la seva inestimable paciència i el seu recolzament durant aquests anys.

I per últim, amb menció especial per la meva dona i el meu fill, Mònica i Eric. Sou les persones que heu fet possible la culminació d'aquesta llarga travessia d'aprenentatge. Gràcies, us estimo moltíssim.

Resum

El present treball fi de carrera de l'Enginyeria Tècnica de Sistemes pretén mostrar les pautes seguides per desenvolupar una aplicació del món real basada en l'arquitectura J2EE.

L'objectiu principal del projecte és la realització de l'anàlisi, el disseny i la implementació d'una aplicació web de comerç electrònic; el nom escollit per aquest lloc web és **Enomarket** i estarà destinat a la venda de vins. L'aplicació està dividida en dos grans blocs: una part pública i una part privada. La part d'accés públic, permet consultar el catàleg de vins, afegir-hi articles al cistell de compra i realitzar-ne comandes. La part privada, permet realitzar la gestió, administració i control del l'espai web.

Per al desenvolupament de l'aplicació es fa ús de l'arquitectura J2EE (*Java 2 Enterprise Edition*) seguint el patró de disseny **MVC**¹. Com a entorn de desenvolupament o **IDE**², s'ha fet servir l'Eclipse 3.5 Galileo amb l'incorporació de plug-ins per tal de facilitar-ne la programació. El **SGBD**³ és Oracle i el servidor web emprat és **JBoss**.

Molts dels conceptes estudiats al llarg dels semestres de l'Enginyeria, són aplicats ara en el projecte: programació orientada a objectes, programació en Java, enginyeria del programari, bases de dades relacionals; i d'altres desconeguts fins ara, són analitzats i estudiats, com són els diferents frameworks disponibles al mercat per desenvolupar aplicacions seguint l'especificació J2EE: com Struts 2, Tiles i Hibernate, per exemple.

Àrea de treball: J2EE.

Paraules clau: Java, J2EE, Struts 2, Tiles, Hibernate, patrons de disseny.

¹ MVC, Model-Vista-Controlador.

² Entorn de desenvolupament integrat (de l'anglès IDE).

³ SGBD, sigles de Sistema Gestor de Base de Dades.

Índex de continguts

ENOMARKET, BOTIGA VIRTUAL DE VINS MITJANÇANT TECNOLOGIA J2EE	1
DEDICATÒRIA I AGRAÏMENTS	2
RESUM.....	3
1 INTRODUCCIÓ	7
1.1 JUSTIFICACIÓ I CONTEXT DEL TFC	7
1.2 OBJECTIUS DEL TFC	7
1.3 ENFOCAMENT I MÈTODE SEGUIT.....	8
1.4 PLANIFICACIÓ DEL PROJECTE.....	9
1.4.1 <i>Calendari.....</i>	9
1.4.2 <i>Fites i distribució de tasques del projecte</i>	10
1.4.3 <i>Planificació temporal i diagrama de Gantt.....</i>	11
1.5 PRODUCTES OBTINGUTS	12
1.6 DESCRIPCIÓ DEL CAPÍTOLS DE LA MEMÒRIA	12
2 DESCRIPCIÓ DEL PROJECTE ENOMARKET	13
3 ANÀLISIS	14
3.1 ACTORS.....	14
3.2 MODEL DE CASOS D'ÚS	15
3.2.1 <i>Funcionalitats de l'actor visitant</i>	15
3.2.2 <i>Funcionalitats de l'actor client.....</i>	16
3.2.3 <i>Funcionalitats de l'actor administrador.....</i>	17
3.3 FITXES DELS CASOS D'ÚS	18
3.3.1 <i>Actor visitant.....</i>	18
3.3.2 <i>Actor client.....</i>	19
3.3.3 <i>Actor administrador.....</i>	23
3.4 PROTOTIPUS	28
4 DISSENY	35
4.1 DISSENY DE CLASSES	35
4.1.1 <i>Identificació de classes d'entitat.....</i>	35
4.1.2 <i>Identificació dels atributs de les classes d'entitat.....</i>	35
4.1.3 <i>Diagrama de classes</i>	36
4.2 DISSENY DE LA BASE DE DADES.....	37
4.2.1 <i>Disseny relacional</i>	37
4.3 ARQUITECTURA DEL PROJECTE	38
4.3.1 <i>Patró de disseny MVC.....</i>	38
4.3.2 <i>Struts 2.....</i>	39
4.3.2.1 <i>Model.....</i>	39
4.3.2.2 <i>Vista.....</i>	42
4.3.2.3 <i>Controlador</i>	42
4.3.3 <i>Tiles.....</i>	42
4.3.4 <i>Hibernate.....</i>	42
4.3.5 <i>DisplayTag</i>	43
4.3.6 <i>Log4j.....</i>	43
5 IMPLEMENTACIÓ.....	44

5.1	EINES DE DESENVOLUPAMENT	44
5.2	DECISIONS D'IMPLEMENTACIÓ	44
5.3	ESTRUCTURA DEL PROJECTE	46
5.4	GUÍA D'INSTAL·LACIÓ I CONFIGURACIÓ DE L'APLICACIÓ	50
5.4.1	<i>Entorn de treball</i>	50
5.4.2	<i>Base de dades</i>	50
5.4.3	<i>Servidor d'aplicacions i contenidor web</i>	51
6	CONCLUSIONS	52
7	GLOSSARI	53
8	BIBLIOGRAFIA	54
9	ANNEXOS	55

Índex de figures

FIGURA 1. FLUX DE TREBALL DE LES DIFERENTS FASES DEL PROCÉS UNIFICAT.....	9
FIGURA 2. DATES CLAU DEL PROJECTE.....	9
FIGURA 3. PLANIFICACIÓ I DISTRIBUCIÓ DE LES TASQUES DEL PROJECTE.....	10
FIGURA 4. DIAGRAMA DE GANTT DEL PROJECTE.....	11
FIGURA 5. DIAGRAMA DELS ACTORS DEL SISTEMA.....	14
FIGURA 6. DIAGRAMA DE CASOS D'ÚS DE L'ACTOR VISITANT.....	15
FIGURA 7. DIAGRAMA DE CASOS D'ÚS DE L'ACTOR CLIENT.....	16
FIGURA 8. DIAGRAMA DE CASOS D'ÚS DE L'ACTOR ADMINISTRADOR.....	17
FIGURA 9. ESQUEMA DEL FORMAT DE PÀGINA.....	28
FIGURA 10. PROTOTIP DE FINESTRA INICIAL DEL SISTEMA.....	29
FIGURA 11. PROTOTIP DE FINESTRA DE REGISTRE DE NOU CLIENT.....	29
FIGURA 12. PROTOTIP DE FINESTRA INICIAL DE CLIENT AUTENTICAT.....	30
FIGURA 13. PROTOTIP DE FINESTRA DE CISTELL DE COMPRA DE CLIENT.....	30
FIGURA 14. PROTOTIP DE FINESTRA DE RECERCA AVANÇADA DE CLIENT.....	31
FIGURA 15. PROTOTIP DE FINESTRA INICIAL DE L'ADMINISTRADOR.....	31
FIGURA 16. PROTOTIP DE FINESTRA DE CONSULTA DE DADES D'UNA COMANDA.....	32
FIGURA 17. PROTOTIP DE FINESTRA DE GESTIÓ DE CLIENTS.....	33
FIGURA 18. PROTOTIP DE FINESTRA DE MODIFICACIÓ DE VI.....	34
FIGURA 19. DIAGRAMA DE CLASSES.....	36
FIGURA 20. DIAGRAMA ENTITAT-RELACIÓ DE LA BASE DE DADES.....	37
FIGURA 21. ESQUEMA DE L'ARQUITECTURA DEL SISTEMA.....	38
FIGURA 22. DIAGRAMA DE L'ARQUITECTURA MVC.....	38
FIGURA 23. ARQUITECTURA DEL FRAMEWORK STRUTS 2.....	39
FIGURA 24. ESTRUCTURA DEL DAO DE CLIENT.....	40
FIGURA 25. ESTRUCTURA DEL DAO DE PRODUCTES.....	41
FIGURA 26. ESTRUCTURA DEL DAO DE COMANDES.....	41
FIGURA 27. DETALL DEL FITXER LOGINACTION-VALIDATION.XML.....	45
FIGURA 28. ESTRUCTURA DE DIRECTORIS DEL PROJECTE WEB.....	46
FIGURA 29. DETALL DEL FITXER STRUTS.XML DEL PROJECTE.....	47
FIGURA 30. DETALL DEL FITXER STRUTS.PROPERTIES DEL PROJECTE.....	47
FIGURA 31. DETALL DEL FITXER WEB.XML DEL PROJECTE.....	48
FIGURA 32. DETALL DEL FITXER HIBERNATE.CFG.XML DEL PROJECTE.....	49
FIGURA 33. DETALL DEL FITXER CLIENT.HBM.XML DEL PROJECTE.....	49

1 Introducció

El treball fi de carrera (TFC) consisteix en realitzar l'anàlisi, el disseny i la implementació d'una aplicació en una interfície web, mitjançant la tecnologia Java i l'arquitectura J2EE.

L'objectiu del present capítol és fer la presentació delimitant l'abast del projecte, enumerant els requeriments i les funcionalitats que tindrà l'aplicació web, així com la distribució de les tasques i la planificació temporal del projecte al llarg del semestre.

1.1 Justificació i context del TFC

El treball fi de carrera és una assignatura pensada per realitzar un projecte de síntesi dels coneixements adquirits en altres assignatures i on han de ser posats en pràctica bona part d'aquests coneixements obtinguts al llarg de l'enginyeria.

D'una banda, l'objectiu acadèmic com a culminació dels estudis de l'Enginyeria, és que s'apliquin i es vegin reflectits els coneixements adquirits en els crèdits d'estudi de programació en llenguatge orientat a objectes Java, la gestió i l'estudi de bases de dades relacionals, i l'enginyeria de programari, entre d'altres.

D'altra banda, com a objectius personals i professionals són els de ser capaç de realitzar un projecte sencer partint des de zero, on estiguin presents la majoria de conceptes de l'àrea d'estudi de la programació actual portats a la pràctica i obtenir els coneixements necessaris de les noves tecnologies de l'estàndard J2EE.

J2EE és una tecnologia molt emprada a l'àmbit professional per la quantitat i la riquesa de solucions que es poden emprar i suposa per a mi un alt aprenentatge dels nous estàndards que incorpora aquesta arquitectura. L'ús d'aquest estàndard, aconsegueix tots els requisits que se li demana a qualsevol plataforma empresarial: escalabilitat, fiabilitat, disponibilitat, i tot això es pot aconseguir mitjançant programari lliure. Al mercat, apareixen cada vegada més eines que fan més senzill i ràpid, el disseny i la construcció d'un entorn web de comerç electrònic.

La principal motivació com a estudiant amb certs coneixements de Java, és dissenyar e implementar un exemple típic d'una aplicació web distribuïda, mitjançant la tecnologia J2EE i seguint els patrons de disseny MVC, sense tenir experiència prèvia en aquesta arquitectura.

1.2 Objectius del TFC

Fa ja uns anys que la indústria del programari està adoptant més i més l'orientació a objectes. Per altra banda, les aplicacions d'empresa, degut a les tendències actuals, estant adoptant cada cop més una estructura distribuïda. Això, afegit a l'imposició àmplia d'Internet a nivell empresarial fa que cada cop més les aplicacions d'empresa usin la xarxa com a vehicle de comunicació bàsic. A més, la possibilitat de fer servir la web com a interfície està fent que moltes aplicacions s'estiguin migrant cap a aquest entorn.

Em l'actualitat han aparegut diverses tecnologies per a facilitar i estandarditzar el desenvolupament d'aplicacions d'aquest entorn; concretament, la tecnologia Java s'ha constituït des de ja fa anys en un puntal fort en aquest aspecte.

Java, com a llenguatge de desenvolupament, juntament amb l'arquitectura J2EE s'ha convertit en un estàndard en el món de la indústria per la realització d'aplicacions empresarials a Internet.

L'objectiu general de l'àrea és aprofundir en l'ús de la tecnologia Java, mitjançant l'arquitectura J2EE, amb la principal finalitat d'obtenir nous coneixements de la plataforma, que implicarà:

- Estudiar els **frameworks**⁴ i estàndards disponibles per a les diferents capes de l'aplicació, tant els definits per l'estàndard J2EE (JavaServer Pages, Servlets o Enterprise JavaBeans), com els desenvolupats per la comunitat amb llicència lliure (WebWork amb Struts 2, JBoss Community amb Hibernate, o Tiles d'Apache).
- Fer el disseny de l'aplicació en tres capes: la capa de presentació, la capa de negoci i la capa de dades, fent servir els patrons de disseny més adequats i implementant dins cada capa, els aspectes corresponents segons la seva distribució.
- Perfilar els aspectes d'usabilitat⁵ i dissenyar una interfície web pràctica i alhora intuïtiva tenint en compte el context web de comerç en línia en el que ens trobem.
- Mantenir separada la lògica de negoci de la persistència de dades, i garantir les transaccions realitzades per diferents usuaris en concurrència.
- Afegir seguretat a l'aplicació mitjançant mecanismes d'autenticació i establint els diferents rols d'usuari.

1.3 Enfocament i mètode seguit

Durant molt de temps s'ha fet ús del model tradicional en cascada, el qual ha demostrat que no reflecteix adequadament la complexitat inherent al procés de desenvolupament de software. Els problemes que representa aquest model neixen de la seva pròpia estructura, al ser una seqüència de grans etapes que requereixen tenir la documentació completa abans de continuar amb la següent etapa.

Per solucionar aquest problema es deu fer ús de mètodes interactius i incrementals, i aquest treball així ho intenta, per tant segueix les fases del Procés Unificat de Desenvolupament de Software, que són les següents a grans trets:

- Especificació dels requeriments de l'aplicació.
- Anàlisi de l'aplicació.
- Disseny de l'arquitectura, model de dades i lògica de negoci.
- Disseny de l'estructura de l'aplicació.
- Implementació.
- Proves i tests de l'aplicació.

Els beneficis d'un procés iteratiu controlat són: la reducció del risc de no treure el producte en el calendari previst, accelerar el ritme de desenvolupament i que s'adapti millor a les necessitats del client:

⁴ Framework, és una estructura conceptual i tecnològica de mòduls de software concrets, en base al qual un altre projecte de software pot ser organitzat i desenvolupat.

⁵ Disseny del producte per que s'ajusti a les necessitats, capacitats, i expectatives de l'usuari.

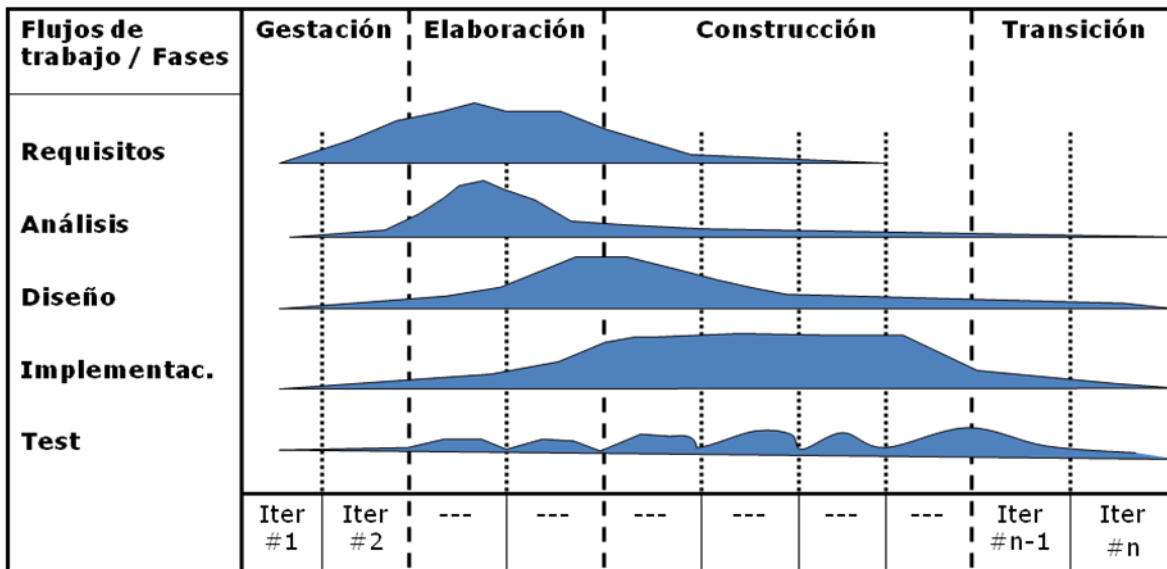


Figura 1. Flux de treball de les diferents fases del procés unificat.

1.4 Planificació del projecte

1.4.1 Calendari

Per la realització del TFC hi ha delimitades unes dates d'entrega parcial de l'evolució del projecte, per tal de ser avaluat el desenvolupament del treball realitzat pel consultor de l'assignatura. Les línies marcades de color vermell són les dates clau del projecte:

Data inici del semestre	04/03/2011
Proposta TFC al consultor	06/03/2011
Lliurament PAC1 (Pla de treball)	16/03/2011
1er Testeig dels builds	25/03/2011
Entrega tests bàsics	30/03/2011
Lliurament PAC2 (Anàlisi i disseny)	14/04/2011
Entrega parcial .war (aplicació web) 80%	20/05/2011
Lliurament PAC3 (Implementació)	23/05/2011
Entrega parcial .war (aplicació web)	14/06/2011
Entrega final del TFC (Memòria i presentació)	20/06/2011

Figura 2. Dates clau del projecte.

1.4.2 Fites i distribució de tasques del projecte

Tenint en compte les restriccions de les dates clau d'entrega de les PAC, i el handicap addicional de falta d'experiència en l'arquitectura J2EE, que fa necessari afegir temps de dedicació a l'estudi teòric i pràctic de les noves tecnologies, aquesta és la distribució temporal de les fites i tasques a realitzar del projecte:

Id	Nombre de tarea	Duración	Comienzo	Fin
1	TFC J2EE ENOMARKET	109 días	vie 04/03/11	lun 20/06/11
2	Fase inicial del projecte	13 días	vie 04/03/11	mié 16/03/11
3	Proposta de TFC	3 días	vie 04/03/11	dom 06/03/11
4	Anàlisi i definició de requeriments	3 días	lun 07/03/11	mié 09/03/11
5	Elaboració del pla de treball	7 días	jue 10/03/11	mié 16/03/11
6	Lliurament Pla de treball (PAC1)	0 días	mié 16/03/11	mié 16/03/11
7	Fase d'anàlisi funcional i disseny tècnic	29 días	jue 17/03/11	jue 14/04/11
8	Recopilació d'informació i pressa de contacte dels frameworks	6 días	jue 17/03/11	mar 22/03/11
9	Struts 2	6 días	jue 17/03/11	mar 22/03/11
10	Tiles i JSP	6 días	jue 17/03/11	mar 22/03/11
11	Hibernate	6 días	jue 17/03/11	mar 22/03/11
12	Preparació de l'entorn de treball	3 días	mié 23/03/11	vie 25/03/11
13	Instal·lació de l'entorn Eclipse	2 días	mié 23/03/11	jue 24/03/11
14	Instal·lació de Oracle BBDD	2 días	mié 23/03/11	jue 24/03/11
15	Configuració de l'entorn de treball	3 días	mié 23/03/11	vie 25/03/11
16	Testeig bàsic	5 días	sáb 26/03/11	mié 30/03/11
17	Accés bàsic a les dades	3 días	sáb 26/03/11	lun 28/03/11
18	Implementació amb MVC	2 días	mar 29/03/11	mié 30/03/11
19	Disseny de l'aplicació	15 días	jue 31/03/11	jue 14/04/11
20	Anàlisi de requeriments	1 día	jue 31/03/11	jue 31/03/11
21	Definició de casos d'ús i actors	5 días	vie 01/04/11	mar 05/04/11
22	Model de pantalles o prototipus	3 días	mar 05/04/11	jue 07/04/11
23	Disseny relacional de la base de dades	4 días	jue 07/04/11	dom 10/04/11
24	Diagrama de classes principal	2 días	lun 11/04/11	mar 12/04/11
25	Diagrama d'arquitectura de l'aplicació	3 días	mar 12/04/11	jue 14/04/11
26	Lliurament PAC2	0 días	jue 14/04/11	jue 14/04/11
27	Fase d'implementació del projecte	39 días	vie 15/04/11	lun 23/05/11
28	Implementació del 80% del codi	25 días	vie 15/04/11	lun 09/05/11
29	Proves unitàries casos d'ús	14 días	mar 10/05/11	lun 23/05/11
30	Lliurament PAC3	0 días	lun 23/05/11	lun 23/05/11
31	Fase final del projecte	28 días	mar 24/05/11	lun 20/06/11
32	Implementació del 20% restant de codi	15 días	mar 24/05/11	mar 07/06/11
33	Proves integrals i tests	6 días	jue 09/06/11	mar 14/06/11
34	Redacció de la memòria	4 días	mié 15/06/11	sáb 18/06/11
35	Elaboració de la presentació	2 días	sáb 18/06/11	lun 20/06/11
36	Lliurament PAC4	0 días	lun 20/06/11	lun 20/06/11

Figura 3. Planificació i distribució de les tasques del projecte.

1.4.3 Planificació temporal i diagrama de Gantt

El diagrama de Gantt és una popular eina gràfica per mostrar el temps de dedicació previst per a les diferents tasques i activitats dins el temps total establert per obtenir un projecte complex.

En següent diagrama de Gantt mostra la planificació temporal de cada tasca i les diferents fites:

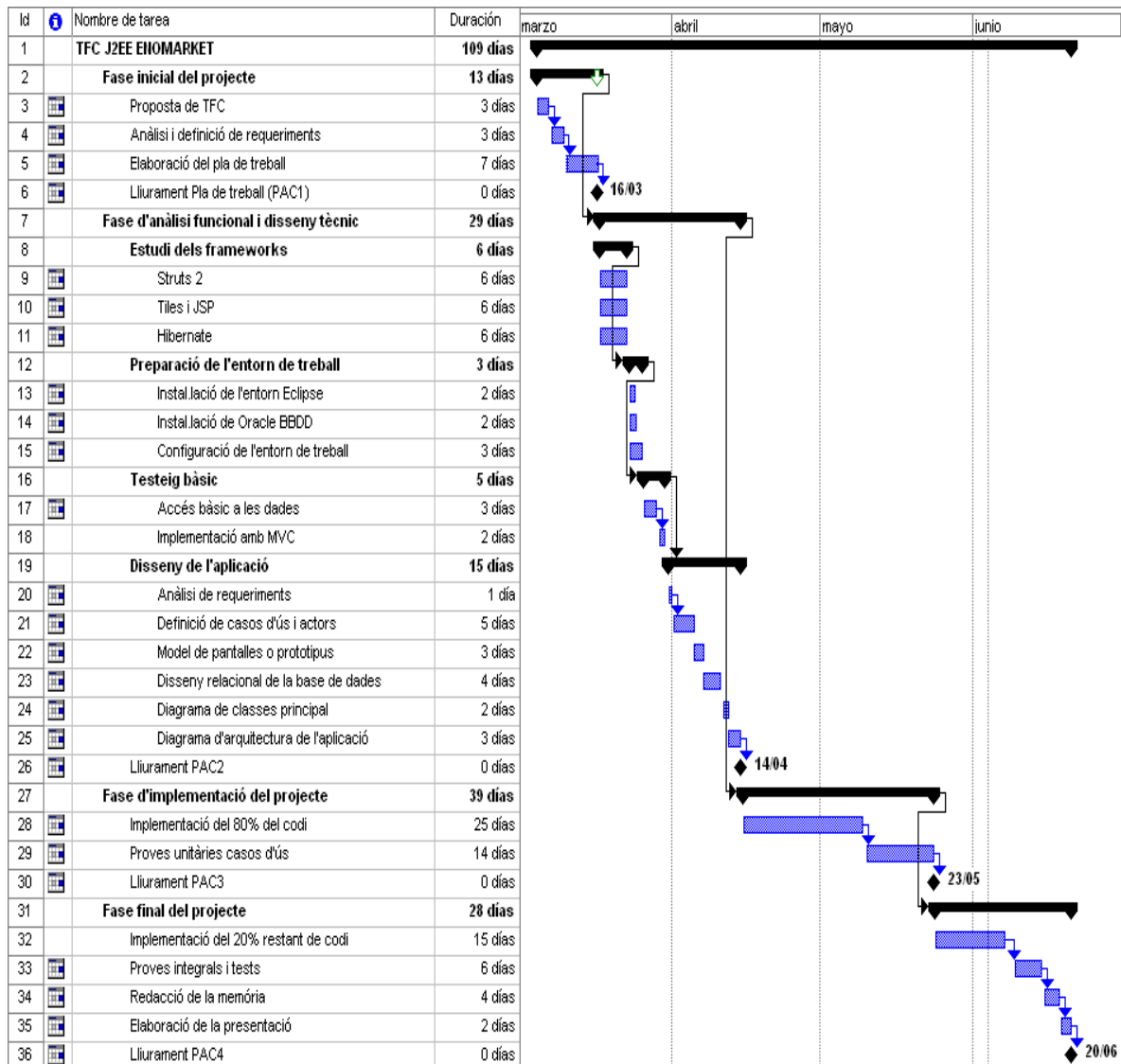


Figura 4. Diagrama de Gantt del projecte.

1.5 Productes obtinguts

El productes obtinguts al finalitzar el TFC són els següents:

- Una **aplicació web**, totalment funcional de venda de vins per Internet, desenvolupada seguint l'especificació J2EE i amb programari lliure.
- La **memòria**, que recull en forma escrita tots els aspectes tècnics del projecte, conté tota la informació rellevant per comprendre el problema plantejat en el TFC, així com la metodologia emprada per la resolució del problema plantejat.
- La **presentació**, que ofereix una perspectiva general del TFC, sintetitzant de forma clara i concisa el treball realitzat i els resultats obtinguts.

1.6 Descripció del capítols de la memòria

A continuació es presenta de forma breu els principals aspectes de la resta de capítols inclosos a la memòria:

- **Descripció del projecte Enomarket:** s'especifica l'abast del projecte final i es delimiten les funcionalitats de l'aplicació.
- **Anàlisis:** s'especifiquen els actors i es detallen els casos d'ús segons els requeriments funcionals de l'aplicació.
- **Disseny:** s'identifiquen les classes d'entitat principals i les seus atributs i es realitza un prototipus de les finestres de l'interfície d'usuari de l'aplicació web. Es presenta també, tant el disseny de la base de dades, com el disseny de l'arquitectura del projecte.
- **Implementació:** es presenten les eines de desenvolupament, així com els resultats obtinguts i les decisions preses durant l'implementació de l'aplicació. S'indiquen tant l'estructura del projecte, com els passos a seguir per a una correcta instal·lació, configuració i execució de l'aplicació.
- **Conclusions:** es presenten les valoracions finals una vegada finalitzada l'implementació i s'enumeren les millores per a pròximes versions.

2 Descripció del projecte Enomarket

Enomarket ha de ser una web dedicada a la venda en línia de vins amb opcions molt semblants a d'altres portals web de comerç electrònic. L'aplicació web, desenvolupada en Java en un entorn client-sevidor amb suport d'un sistema de base de dades, disposarà d'una part d'accés pública – entorn per a convidats i usuaris registrats -, i una part d'accés privada, per a la gestió i l'administració de les dades de l'espai web.

Les principals funcionalitats i requeriments del projecte a l'àrea pública són els següents:

- Poder registrar-se com a client amb les seves dades personals i posteriorment identificar-se com a tal a la web.
- Una vegada registrat permet fer la modificació de les seves dades.
- Poder consultar el catàleg de vins de cada denominació d'origen a través del menú lateral.
- Poder fer recerques al catàleg de vins existent.
- Tenir accés al cistell de compra de client, on pot confirmar la comanda o fer-hi modificacions dels articles afegits.
- Poder desconnectar-se com a client de la web.

A la part privada, prèvia autenticació d'un usuari predefinit com administrador, les funcionalitats i requeriments són els següents:

- Donar d'alta, modificar i eliminar denominacions d'origen al catàleg.
- Donar d'alta, modificar i eliminar els vins al catàleg.
- Gestionar els usuaris de la base de dades: permetent a l'administrador fer modificacions a les dades personals i donant de baixa usuaris registrats.
- Poder desconnectar-se com a usuari administrador de la web.

3 Anàlisis

3.1 Actors

Els actors són totes les entitats externes al sistema que guarden una relació amb aquest i que li demanden una funcionalitat. Poden ser operadors humans, o fins i tot entitats abstractes, com el temps. En el cas dels sers humans, es poden veure als actors com a definicions de rol, un mateix individu pot correspondre a un o més actors. Aquests són els actors que interactuen en el sistema:

- **Visitant:** és l'usuari anònim que pot consultar el catàleg sense haver-ne autenticat l'accés.
- **Client:** és l'usuari registrat al sistema, que una vegada autenticat, amplia les funcionalitats de l'usuari anònim: fer recerques avançades, pot realitzar comandes mitjançant el cistell de compra i consultar-ne les realitzades i modificar les seves dades personals al sistema.
- **Administrador:** és l'usuari autenticat al sistema amb les credencials d'administrador. Té l'autorització per encarregar-se de la gestió, administració i manteniment de les dades del sistema, com són les dades d'usuari, vins, denominacions i comandes.

Es pot establir una única relació entre actors que és la relació de generalització, es pot veure que l'actor client és una especialització o extensió de l'actor visitant, ja que comparteix i amplia les descripcions específiques d'aquest últim.

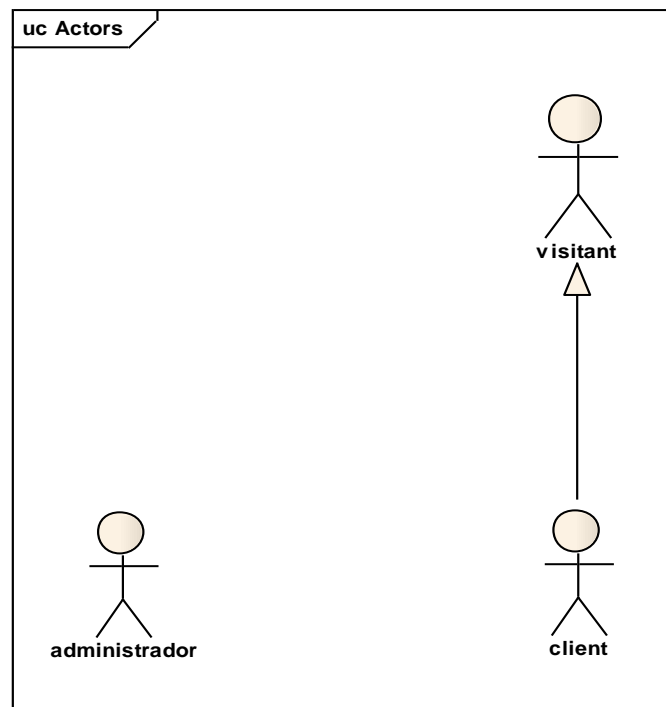


Figura 5. Diagrama dels actors del sistema.

3.2 Model de casos d'ús

Els casos d'ús permeten capturar el comportament desitjat del sistema sense haver d'especificar com s'implementa. A continuació s'enumeren tots els requeriments específics de cada actor i es mostra el diagrama dels casos d'ús corresponents a cadascun. Aquest diagrama és una de les eines **UML**⁶ més comuns per validar l'arquitectura i verificar el sistema en el transcurs del desenvolupament d'aquest.

Aquestes són les funcionalitats del sistema:

3.2.1 Funcionalitats de l'actor visitant

- Consultar el catàleg de vins segons les denominacions existents.
- Consultar les dades de les denominacions existents.
- Registrar-se com a client al sistema.
- Realitzar recerques de vins.
- Veure les fitxes amb les dades dels vins existents.

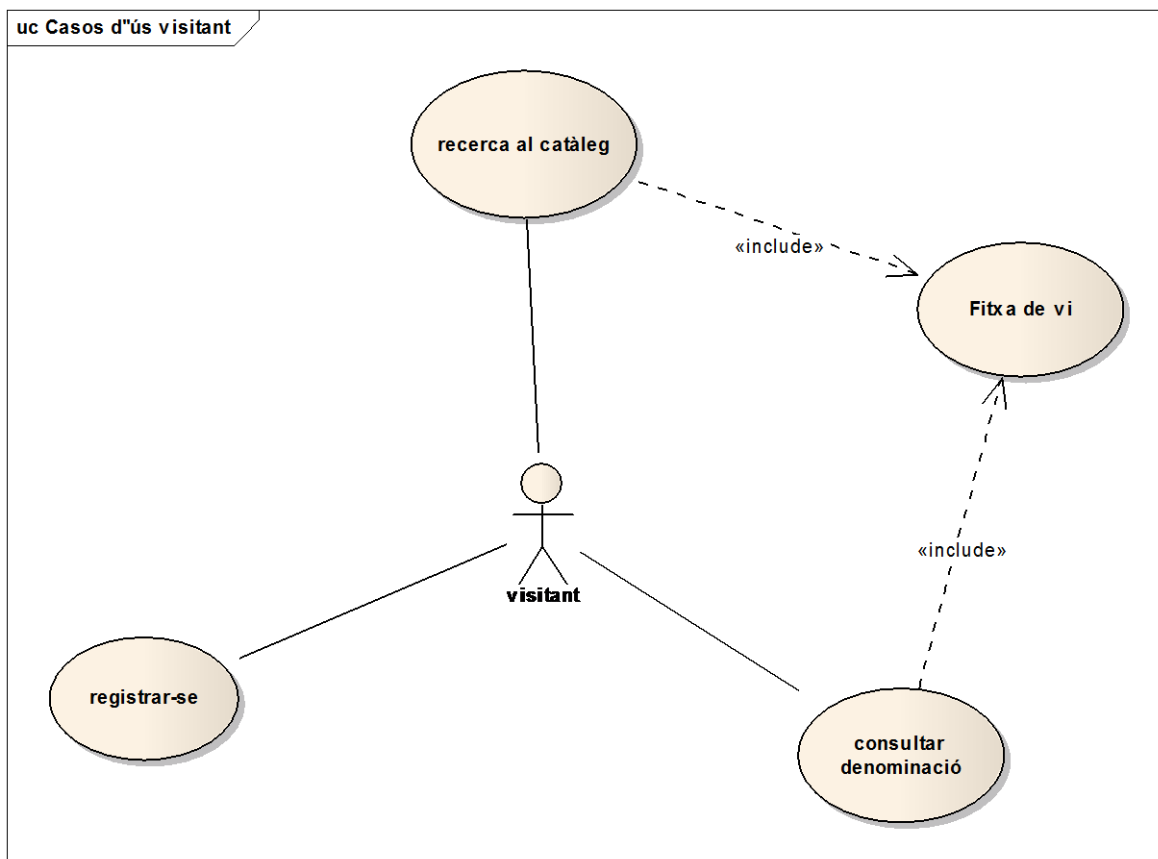


Figura 6. Diagrama de casos d'ús de l'actor visitant.

⁶ UML és un conjunt d'eines, que permeten analitzar i dissenyar sistemes orientats a objectes,

3.2.2 Funcionalitats de l'actor client

- Consultar el catàleg de vins segons les denominacions existents.
- Realitzar recerques de vins.
- Realitzar recerques avançades de vins segons els criteris existents.
- Veure les fitxes amb les dades dels vins existents.
- Identificar-se a la web amb les credencials de client.
- Consultar les seves dades personals i fer-hi modificacions.
- Gestionar el seu cistell de compra.
- Realitzar comandes dels vins del cistell de compra.
- Consultar les comandes que ha realitzat.

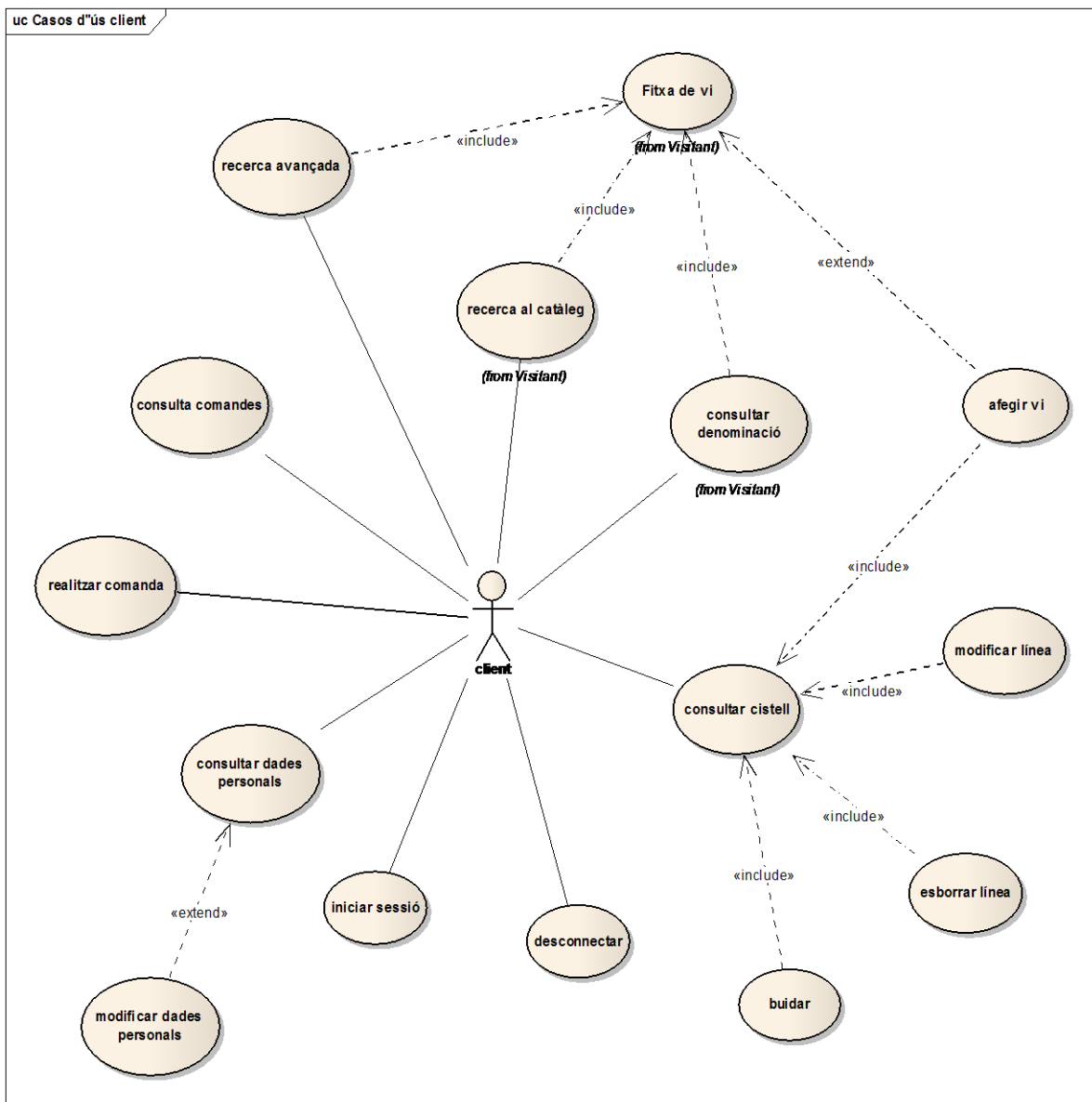


Figura 7. Diagrama de casos d'ús de l'actor client.

3.2.3 Funcionalitats de l'actor administrador

- Identificar-se a la web.
- Consultar totes les dades de les comandes realitzades al sistema.
- Gestionar els vins (alta, modificació i baixa).
- Gestionar les denominacions d'origen (alta, modificació i baixa).
- Gestionar els clients (alta, modificació i baixa).

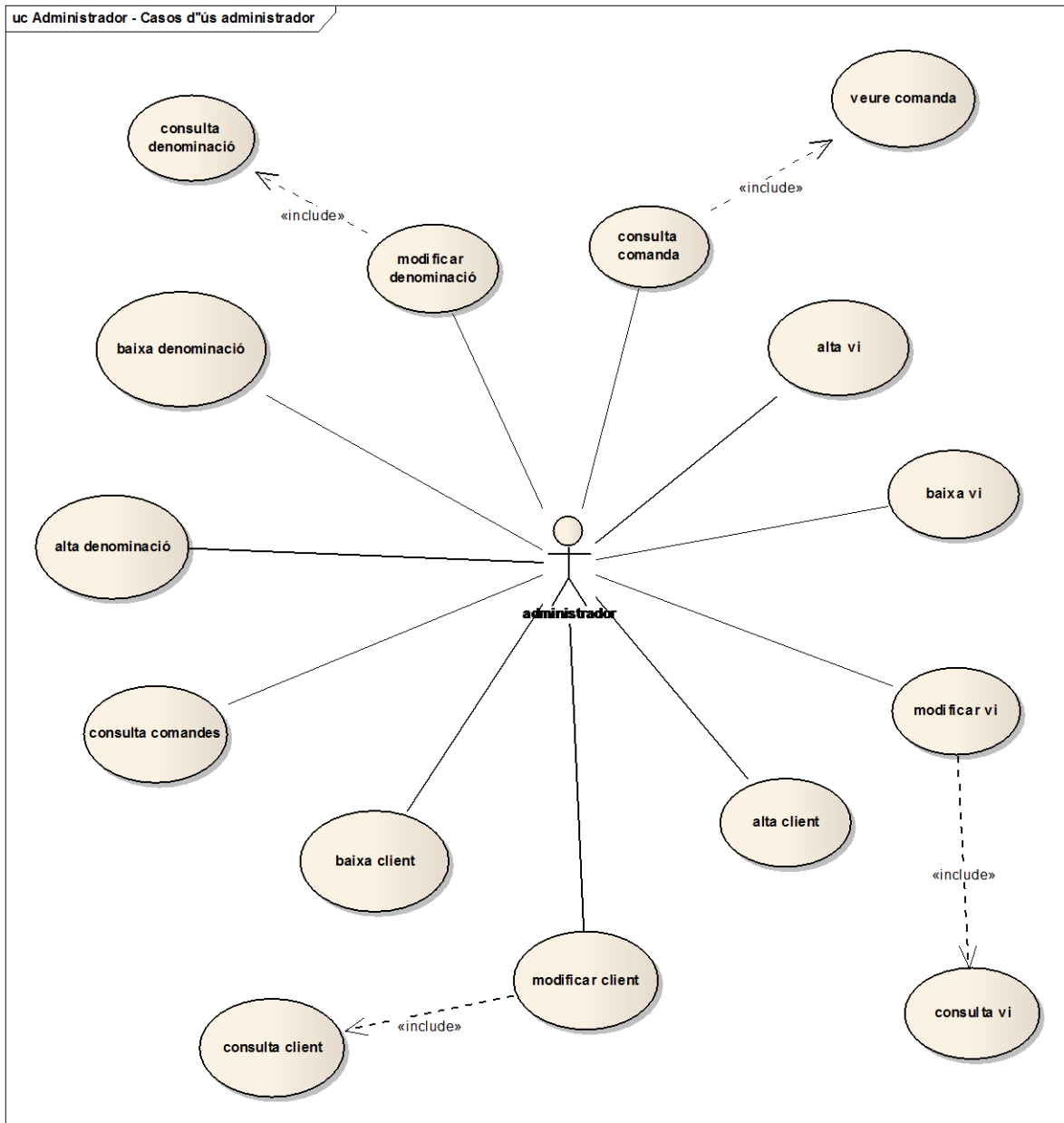


Figura 8. Diagrama de casos d'ús de l'actor administrador.

3.3 Fitxes dels casos d'ús

3.3.1 Actor visitant

Cas d'ús		Registrar-se
Resum de la funcionalitat	Dona d'alta el visitant com a client	
Actors	Visitant	
Casos relacionats	Cap	
Precondició	Cap	
Postcondició	S'afegeix un nou client amb les dades introduïdes al sistema	
Procés normal principal		
<ol style="list-style-type: none"> 1. El sistema sol·licita les dades a l'actor visitant 2. El visitant omple les dades demanades 3. El sistema comprova que existeix com a client i valida que les dades siguin correctes (dades imprescindibles i sense errors) 4. El sistema l'enregistra com a client 		
Alternatives de procés i alternatives		
<ol style="list-style-type: none"> 3. El client existeix al sistema o falten dades necessàries <ol style="list-style-type: none"> a. El sistema torna al procés 1 		

Cas d'ús		Consultar denominació
Resum de la funcionalitat	Mostra els vins d'una denominació d'origen	
Actors	Visitant i Client	
Casos relacionats	Fitxa de vi	
Precondició	Cap	
Postcondició	Es mostra el llistat de vins de la denominació d'origen seleccionada	
Procés normal principal		
<ol style="list-style-type: none"> 1. L'actor selecciona una denominació d'origen de la llista que es presenta al sistema 2. El sistema mostra el llistat de vins de la selecció realitzada 		
Alternatives de procés i alternatives		
<ol style="list-style-type: none"> 2. No hi ha vins de la denominació seleccionada <ol style="list-style-type: none"> a. El sistema presenta la llista buida b. El sistema finalitza el cas d'ús 		

Cas d'ús		Recerca al catàleg
Resum de la funcionalitat	Llistat dels vins trobats pel criteri de recerca introduït al quadre de text del menú superior	
Actors	Visitant i Client	
Casos relacionats	Consultar denominació	
Precondició	Cap	

Postcondició	Es mostra el llistat de vins trobats coincidents amb les dades introduïdes a través del quadre de text
Procés normal principal	
<ol style="list-style-type: none"> 1. L'actor introdueix el text que vol trobar al catàleg 2. El sistema busca totes les referències d'aquest text al atribut NOM del catàleg de vins existent. 3. El sistema mostra el llistat de vins trobats coincidents amb la paraula/es introduïdes 	
Alternatives de procés i alternatives	
<ol style="list-style-type: none"> 2. El sistema no troba cap coincidència amb el criteri introduït <ol style="list-style-type: none"> a. El sistema mostra la llista buida b. El sistema finalitza el cas d'ús 	

Cas d'ús	Fitxa de vi
Resum de la funcionalitat	Es mostren les característiques d'un vi
Actors	Visitant i Client
Casos relacionats	Consultar denominació, recerca al catàleg i afegir vi
Precondició	Cap
Postcondició	Es mostra la fitxa completa d'un vi afegint components d'utilitat segons el tipus d'usuari que està consultant el sistema
Procés normal principal	
<ol style="list-style-type: none"> 1. El sistema mostra totes les dades del vi seleccionat 	
Alternatives de procés i alternatives	
Cap	

3.3.2 Actor client

Cas d'ús	Iniciar sessió
Resum de la funcionalitat	Un usuari intenta fer login
Actors	Client
Casos relacionats	Cap
Precondició	Usuari està donat d'alta en el sistema
Postcondició	L'usuari queda autenticat com a client i es mostra el seu perfil
Procés normal principal	
<ol style="list-style-type: none"> 1. L'actor introdueix les dades d'usuari i contrasenya al sistema 2. El sistema autentica les dades introduïdes i reconeix l'usuari 3. El sistema mostra el perfil web del client 	
Alternatives de procés i alternatives	
<ol style="list-style-type: none"> 2. El sistema no autentica les dades introduïdes <ol style="list-style-type: none"> a. El sistema mostra un missatge d'avís de login incorrecte b. El sistema finalitza el cas d'ús 	

Cas d'ús		Desconnectar
Resum de la funcionalitat	L'usuari autenticat tanca la sessió al sistema	
Actors	Client i Administrador	
Casos relacionats	Cap	
Precondició	Usuari ha iniciat sessió en el sistema	
Postcondició	L'usuari deixa d'estar autenticat al sistema	
Procés normal principal		
1. El sistema allibera i tanca la sessió de l'usuari autenticat		
Alternatives de procés i alternatives		
Cap		

Cas d'ús		Consultar dades personals
Resum de la funcionalitat	L'usuari autenticat pot consultar les seves dades personals	
Actors	Client	
Casos relacionats	Cap	
Precondició	Usuari ha iniciat sessió en el sistema	
Postcondició	Es mostren les dades personals de l'usuari de la sessió	
Procés normal principal		
1. El sistema mostra les dades registrades de l'usuari		
2. <i>Inclusió modificar dades personals</i>		
Alternatives de procés i alternatives		
Cap		

Cas d'ús		Modificar dades personals
Resum de la funcionalitat	L'usuari autenticat pot modificar les seves dades personals	
Actors	Client	
Casos relacionats	Cap	
Precondició	Usuari ha iniciat sessió en el sistema	
Postcondició	L'usuari de la sessió modifica les dades personals	
Procés normal principal		
1. L'usuari introdueix les noves dades personals		
2. El sistema valida les dades introduïdes		
3. El sistema registra els canvis realitzats al sistema		
Alternatives de procés i alternatives		
2. El sistema no valida les dades introduïdes		
a. El sistema mostra un missatge d'error		
b. El sistema torna al procés 1.		

Cas d'ús		Consultar cistell
Resum de la funcionalitat		Es consulta l'estat actual del cistell de compra
Actors		Client
Casos relacionats		Buidar cistell i esborrar línia
Precondició		Usuari ha iniciat sessió en el sistema
Postcondició		Es mostra el contingut actual del cistell de compra
Procés normal principal		
1. El sistema mostra els vins afegits al cistell, les quantitats, els preus i el cost total		
Alternatives de procés i alternatives		
3. El cistell es buit <ol style="list-style-type: none">El sistema mostra el cistell de compra buitEl sistema finalitza el cas d'ús		

Cas d'ús		Esborrar línia
Resum de la funcionalitat		S'elimina una línia del cistell de compra
Actors		Client
Casos relacionats		Realitzar comanda
Precondició		Usuari ha iniciat sessió en el sistema i el cistell no es buit
Postcondició		S'elimina el vi del cistell de compra
Procés normal principal		
1. L'usuari selecciona la línia a esborrar del cistell		
2. El sistema actualitza el cistell de compra		
3. <i>Inclusió consultar cistell</i>		
Alternatives de procés i alternatives		
Cap		

Cas d'ús		Buidar cistell
Resum de la funcionalitat		Es buida per complet el cistell de compra
Actors		Client
Casos relacionats		Consultar cistell
Precondició		Usuari ha iniciat sessió en el sistema i el cistell no es buit
Postcondició		S'eliminen totes les línies del cistell de compra
Procés normal principal		
1. El sistema elimina el contingut del cistell de compra		
2. El sistema actualitza el cistell de compra		
3. <i>Inclusió consultar cistell</i>		
Alternatives de procés i alternatives		
Cap		

Cas d'ús	Afegir vi
Resum de la funcionalitat	Permet afegir un vi al cistell de compra
Actors	Client
Casos relacionats	Consultar cistell i llistar vins
Precondició	Usuari ha iniciat sessió en el sistema
Postcondició	S'afegeix el vi i la quantitat introduïda al cistell de compra
Procés normal principal	
<ol style="list-style-type: none"> 1. <i>Extensió Fitxa de vi</i> 2. L'usuari selecciona el vi i introdueix la quantitat d'ampolles que desitja 3. El sistema valida la quantitat introduïda 4. <i>Inclusió consultar cistell</i> 	
Alternatives de procés i alternatives	
<ol style="list-style-type: none"> 4. La quantitat introduïda és un enter menor a 0 o superior a 24 <ol style="list-style-type: none"> a. El sistema torna a la fitxa del vi consultat b. El sistema finalitza el cas d'ús 	

Cas d'ús	Realitzar comanda
Resum de la funcionalitat	Es confirma la compra dels vins del cistell de compra
Actors	Client
Casos relacionats	Cap
Precondició	Usuari ha iniciat sessió en el sistema
Postcondició	El sistema enregistra una nova comanda al sistema amb els vins del cistell de compra
Procés normal principal	
<ol style="list-style-type: none"> 1. El sistema valida el cistell de compra 2. El sistema enregistra la nova comanda 	
Alternatives de procés i alternatives	
<ol style="list-style-type: none"> 5. El cistell es buit <ol style="list-style-type: none"> a. El sistema mostra el cistell de compra buit b. El sistema finalitza el cas d'ús 	

3.3.3 Actor administrador

Cas d'ús	Alta client
Resum de la funcionalitat	Creació d'un nou client al sistema
Actors	Administrador
Casos relacionats	Cap
Precondició	Usuari administrador ha iniciat sessió en el sistema
Postcondició	El nou client queda enregistrat al sistema
Procés normal principal <ol style="list-style-type: none">1. El sistema sol·licita les dades del nou usuari2. L'administrador introdueix les dades3. El sistema valida les dades introduïdes4. El sistema enregistra el nou client	
Alternatives de procés i alternatives <ol style="list-style-type: none">3. El sistema no valida les dades, usuari ja existent o falten dades<ol style="list-style-type: none">a. El sistema mostra un missatge d'errorb. El sistema torna al procés 2.	

Cas d'ús	Consulta client
Resum de la funcionalitat	Consulta les dades d'un client del sistema
Actors	Administrador
Casos relacionats	Modificar client
Precondició	Usuari administrador ha iniciat sessió en el sistema
Postcondició	Consulta les dades del client al sistema
Procés normal principal <ol style="list-style-type: none">1. El sistema mostra la llista de clients existents	
Alternatives de procés i alternatives <ol style="list-style-type: none">1. L'administrador vol fer una consulta avançada<ol style="list-style-type: none">a. El sistema sol·licita els criteris de recercab. L'administrador introdueix els criterisc. El sistema mostra els clients que satisfan les restriccionsd. El sistema torna al procés 2	

Cas d'ús	Baixa client
Resum de la funcionalitat	Eliminació d'un client del sistema
Actors	Administrador
Casos relacionats	Cap
Precondició	Usuari administrador ha iniciat sessió en el sistema i el client existeix al sistema
Postcondició	El client és eliminat del sistema
Procés normal principal	

1. El sistema mostra la llista de clients existents
2. L'administrador selecciona el client
3. El sistema elimina el client

Alternatives de procés i alternatives

Cap

Cas d'ús	Modificar client
Resum de la funcionalitat	Modifica les dades d'un client del sistema
Actors	Administrador
Casos relacionats	Consulta client
Precondició	Usuari administrador ha iniciat sessió en el sistema i el client existeix al sistema
Postcondició	Les dades del client són modificades al sistema
Procés normal principal	
<ol style="list-style-type: none"> 1. <i>Inclusió consulta client</i> 2. L'administrador modifica les dades del client 3. El sistema valida les dades introduïdes 4. El sistema enregistra els canvis al sistema 	
Alternatives de procés i alternatives	
<ol style="list-style-type: none"> 4. L'administrador no confirma <ol style="list-style-type: none"> a. El sistema finalitza el cas d'ús 	

Cas d'ús	Alta denominació
Resum de la funcionalitat	Creació d'una nova denominació d'origen al sistema
Actors	Administrador
Casos relacionats	Cap
Precondició	Usuari administrador ha iniciat sessió en el sistema
Postcondició	La nova denominació d'origen queda registrada al sistema
Procés normal principal	
<ol style="list-style-type: none"> 1. El sistema sol·licita les dades de la nova denominació d'origen 2. L'administrador introdueix les dades 3. El sistema valida les dades introduïdes 4. El sistema registra la nova denominació d'origen 	
Alternatives de procés i alternatives	
<ol style="list-style-type: none"> 3. El sistema no valida les dades, denominació ja existent o falten dades <ol style="list-style-type: none"> a. El sistema mostra un missatge d'error b. El sistema torna al procés 2 	

Cas d'ús	Consulta denominació
Resum de la funcionalitat	Consulta les dades d'una denominació d'origen del sistema
Actors	Administrador
Casos relacionats	Modificar denominació
Precondició	Usuari administrador ha iniciat sessió en el sistema i la denominació existeix al sistema
Postcondició	Consulta les dades de la denominació d'origen al sistema
Procés normal principal	
1. El sistema mostra la llista de denominacions existents	
Alternatives de procés i alternatives	
1. L'administrador vol fer una consulta avançada <ol style="list-style-type: none"> a. El sistema sol·licita els criteris de recerca b. L'administrador introdueix els criteris c. El sistema mostra les denominacions que satisfan les restriccions d. El sistema torna al procés 2 	

Cas d'ús	Baixa denominació
Resum de la funcionalitat	Eliminació d'una denominació d'origen del sistema
Actors	Administrador
Casos relacionats	Cap
Precondició	Usuari administrador ha iniciat sessió en el sistema i la denominació existeix al sistema
Postcondició	La denominació és eliminada del sistema
Procés normal principal	
1. El sistema mostra la llista de denominacions existents 2. L'administrador selecciona la denominació d'origen 3. El sistema elimina la denominació del sistema	
Alternatives de procés i alternatives	
Cap	

Cas d'ús	Modificar denominació
Resum de la funcionalitat	Modifica les dades d'una denominació d'origen del sistema
Actors	Administrador
Casos relacionats	Consulta denominació
Precondició	Usuari administrador ha iniciat sessió en el sistema i la denominació existeix al sistema
Postcondició	Les dades de la denominació són modificades al sistema
Procés normal principal	
1. <i>Inclusió consulta denominació</i>	

2. L'administrador modifica les dades de la denominació
3. El sistema valida les dades introduïdes
4. El sistema enregistra els canvis al sistema

Alternatives de procés i alternatives

4. L'administrador no confirma
 - a. El sistema finalitza el cas d'ús

Cas d'ús	Alta vi
Resum de la funcionalitat	Creació d'un nou vi al sistema
Actors	Administrador
Casos relacionats	Cap
Precondició	Usuari administrador ha iniciat sessió en el sistema
Postcondició	El nou vi queda enregistrat al sistema
Procés normal principal	
<ol style="list-style-type: none"> 1. El sistema sol·licita les dades del nou vi 2. L'administrador introdueix les dades 3. El sistema valida les dades introduïdes 4. El sistema registra el nou vi 	
Alternatives de procés i alternatives	
<ol style="list-style-type: none"> 3. El sistema no valida les dades, vi ja existent o falten dades <ol style="list-style-type: none"> a. El sistema mostra un missatge d'error b. El sistema torna al procés 2 	

Cas d'ús	Consulta vi
Resum de la funcionalitat	Consulta les dades d'un vi del sistema
Actors	Administrador
Casos relacionats	Modificar vi
Precondició	Usuari administrador ha iniciat sessió en el sistema i el vi existeix al sistema
Postcondició	Consulta les dades del vi al sistema
Procés normal principal	
<ol style="list-style-type: none"> 1. El sistema mostra la llista de vins existents 	
Alternatives de procés i alternatives	
<ol style="list-style-type: none"> 1. L'administrador vol fer una consulta avançada <ol style="list-style-type: none"> a. El sistema sol·licita els criteris de recerca b. L'administrador introdueix els criteris c. El sistema mostra els vins que satisfan les restriccions d. El sistema torna al procés 2 	

Cas d'ús	Modificar vi
Resum de la funcionalitat	Modifica les dades d'un vi del sistema
Actors	Administrador
Casos relacionats	Consulta vi
Precondició	Usuari administrador ha iniciat sessió en el sistema i el vi existeix al sistema
Postcondició	Les dades del vi són modificades al sistema
Procés normal principal	
<ol style="list-style-type: none"> 1. <i>Inclusió consulta vi</i> 2. L'administrador modifica les dades del vi 3. El sistema valida les dades introduïdes 4. El sistema enregistra els canvis del vi al sistema 	
Alternatives de procés i alternatives	
<ol style="list-style-type: none"> 4. L'administrador no confirma <ol style="list-style-type: none"> a. El sistema finalitza el cas d'ús 	

Cas d'ús	Baixa vi
Resum de la funcionalitat	Eliminació d'un vi del sistema
Actors	Administrador
Casos relacionats	Cap
Precondició	Usuari administrador ha iniciat sessió en el sistema i el vi existeix al sistema
Postcondició	El vi és eliminat del sistema
Procés normal principal	
<ol style="list-style-type: none"> 1. El sistema mostra la llista de vins existents 2. L'administrador selecciona el vi 3. El sistema elimina el vi 	
Alternatives de procés i alternatives	
Cap	

Cas d'ús	Veure comanda
Resum de la funcionalitat	Es mostren les dades d'una comanda
Actors	Administrador
Casos relacionats	Consulta comanda
Precondició	Usuari administrador ha iniciat sessió en el sistema i la comanda existeix al sistema
Postcondició	Es mostren les dades del client i les línies de compra
Procés normal principal	
<ol style="list-style-type: none"> 1. El sistema mostra les dades de la comanda seleccionada 	
Alternatives de procés i alternatives	

Cap

Cas d'ús	Consulta comanda
Resum de la funcionalitat	Es mostren les comandes del sistema
Actors	Administrador
Casos relacionats	Veure comanda
Precondició	Usuari administrador ha iniciat sessió en el sistema i la comanda existeix al sistema
Postcondició	Es mostren les comandes del sistema
Procés normal principal <ol style="list-style-type: none">1. El sistema mostra la llista de totes les comandes existents2. L'administrador selecciona la comanda desitjada3. <i>Inclusió veure comanda</i>	
Alternatives de procés i alternatives <ol style="list-style-type: none">4. L'administrador vol fer una recerca avançada<ol style="list-style-type: none">a. El sistema sol·licita els criteris de recercab. L'administrador introdueix el número de comanda a trobarc. <i>Inclusió veure comanda</i>	

3.4 Prototipus

Una de les activitats fonamentals durant el Disseny de les interfícies d'usuari es poder representar ràpidament com serà l'interacció amb el sistema, per això s'utilitza la tècnica del prototip, on es representen els principals components del sistema i es validen els requeriments essencials.

L'interfície d'usuari té dos dissenys diferenciats, un disseny per als usuaris i un altre per a l'administrador del sistema. La pàgina conté una capçalera, un menú, el peu de la pàgina i el cos. El format és el següent:

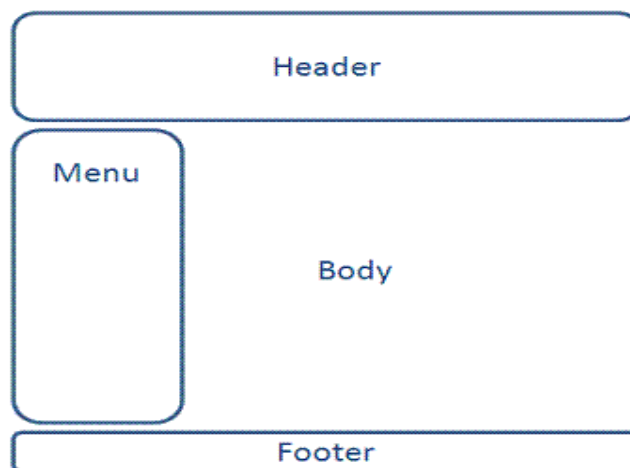


Figura 9. Esquema del format de pàgina.



Figura 10. Prototip de finestra inicial del sistema.



Figura 11. Prototip de finestra de registre de nou client.



Figura 12. Prototip de finestra inicial de client autenticat.



Figura 13. Prototip de finestra de cistell de compra de client.



Figura 14. Prototip de finestra de recerca avançada de client.

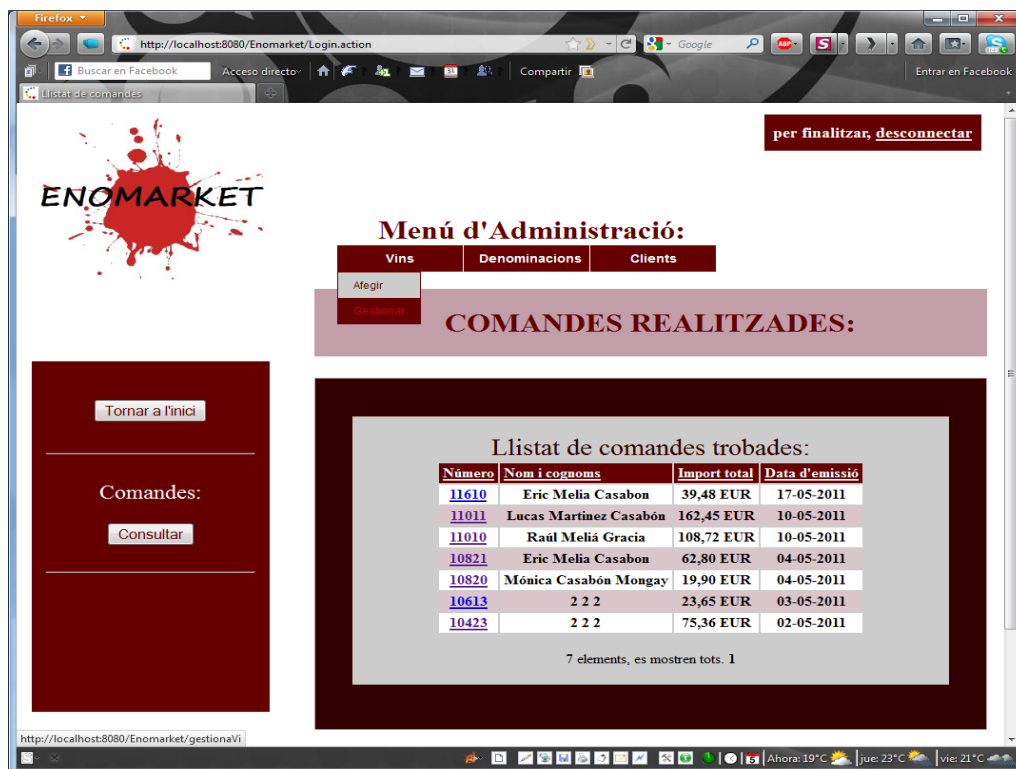


Figura 15. Prototip de finestra inicial de l'administrador.

El següent prototip és la finestra que es mostra al fer la consulta d'una comanda, tant al perfil de client, com al perfil d'administrador. En aquest últim, es mostra un llistat de totes les comandes fent clic al botó del menú lateral de l'esquerra de la pàgina, a continuació seleccionant el número de comanda, es poden veure les dades de la que ha estat seleccionada.

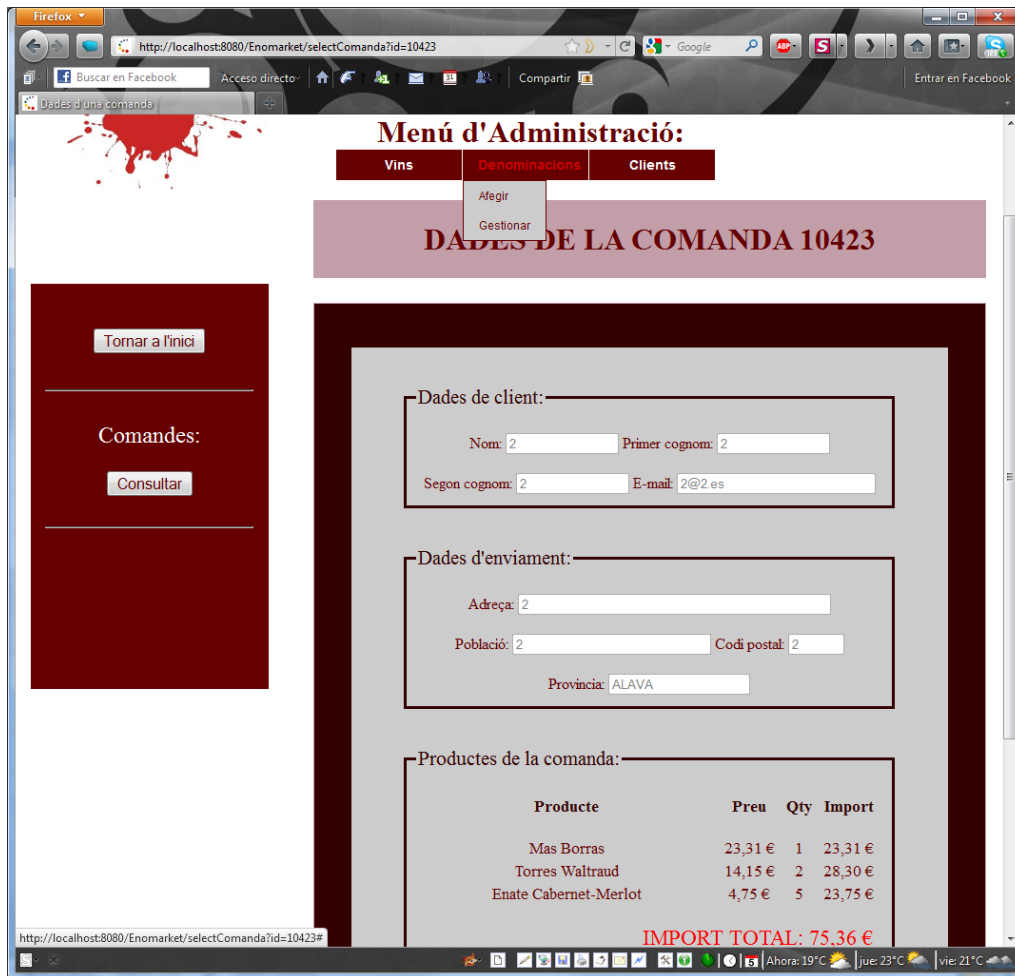


Figura 16. Prototip de finestra de consulta de dades d'una comanda.

La següent figura mostra el prototip de finestra de gestió de clients, que es homogènia per al manteniment de la resta de dades del sistema, com són la gestió de vins i la de denominacions d'origen. Des d'aquestes finestres es poden realitzar modificacions en les dades o bé, donar-les de baixa del sistema.

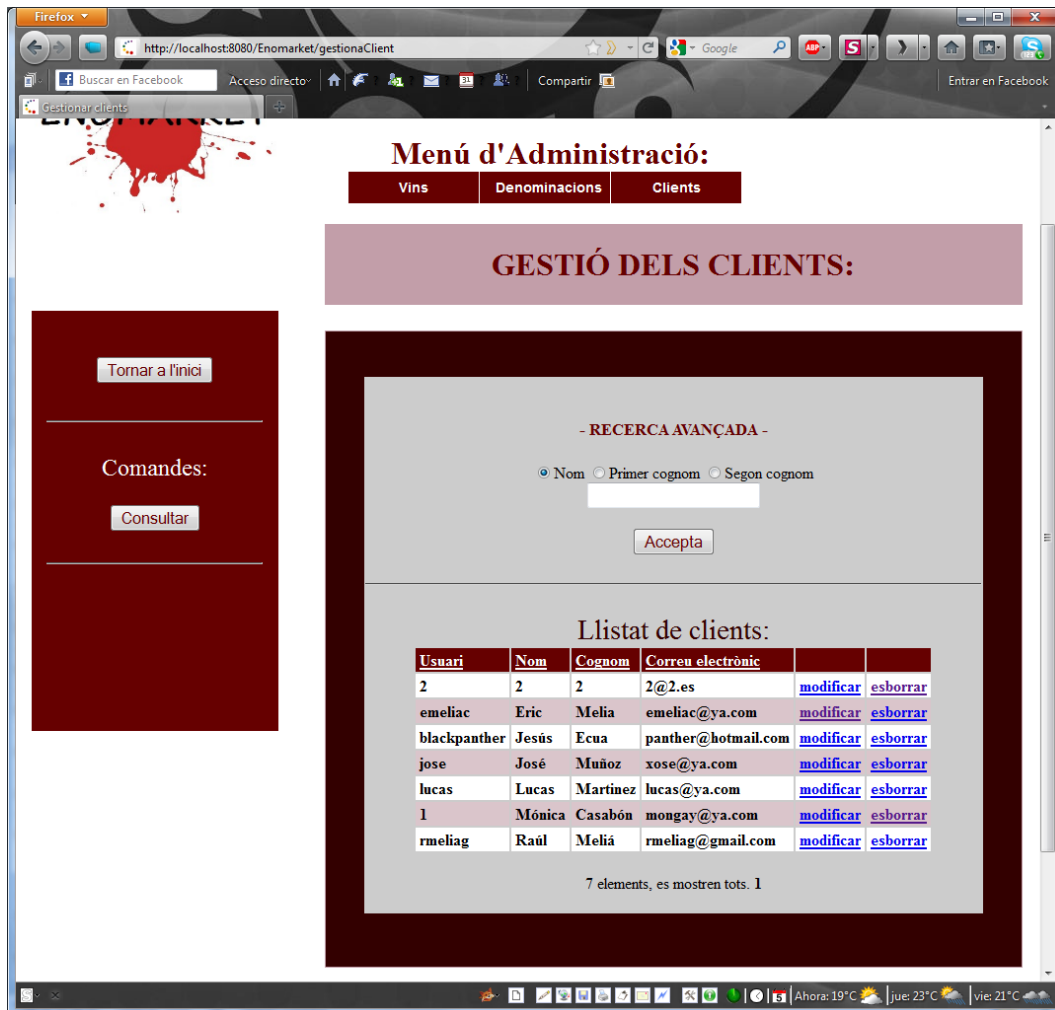


Figura 17. Prototip de finestra de gestió de clients.

La següent figura mostra el prototip de finestra de modificació de vi, que es molt semblant a les altres finestres de modificació de dades. En aquest cas particular, és requisit indispensable omplir tots els camps de les dades que es demanen:

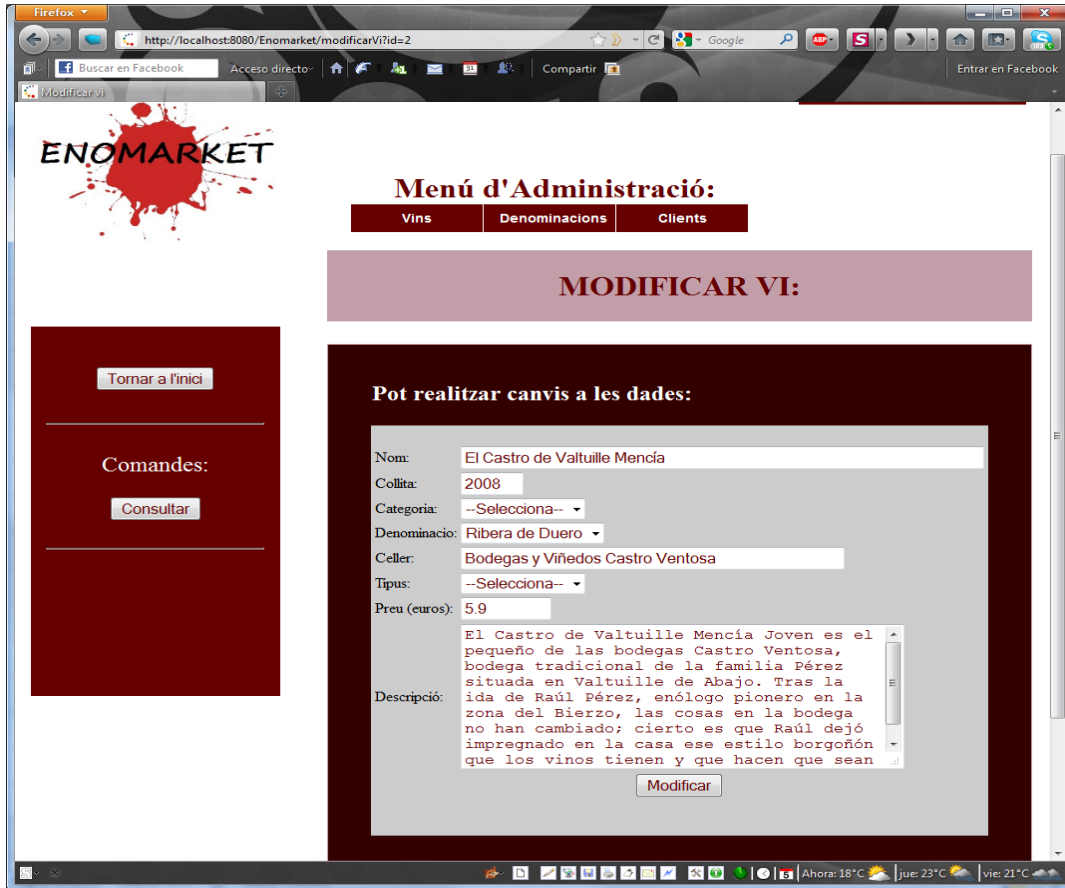


Figura 18. Prototip de finestra de modificació de vi.

4 Disseny

Una vegada acabada la fase d'anàlisi del projecte Enomarket, continuem amb el disseny de l'aplicació mostrant el procés de disseny de classes, el disseny del model relacional de la base de dades i l'arquitectura que es farà servir.

4.1 Disseny de classes

4.1.1 Identificació de classes d'entitat

L'identificació de les classes d'entitat ens permet avaluar les primeres classes on es poden especificar els casos d'ús en forma d'iteracions:

Cas d'ús 1 "registrar-se", classes: visitant.

Cas d'ús 2 "consultar denominació", classes: visitant, client *, vi, denominació.

Cas d'ús 3 "recerca al catàleg", classes: visitant, client *, vi.

Cas d'ús 4 "Fitxa de vi", classes: visitant, client *, vi.

Cas d'ús 5 "iniciar sessió", classes: client, administrador *.

Cas d'ús 6 "desconnectar", classes: client, administrador *.

Cas d'ús 7 "consultar dades personals", classes: client.

Cas d'ús 8 "modificar dades personals", classes: client.

Cas d'ús 9 "consultar cistell", classes: client, vi, cistell.

Cas d'ús 10 "esborrar línia", classes: client, vi, cistell.

Cas d'ús 11 "modificar línia", classes: client, vi, cistell.

Cas d'ús 12 "buidar", classes: client, cistell.

Cas d'ús 13 "afegir vi", classes: client, vi, cistell.

Cas d'ús 14 "realitzar comanda", classes: client, cistell, vi, línia_comanda, comanda.

Cas d'ús 15 "alta client", classes: client *, administrador.

Cas d'ús 16 "consulta client", classes: client *, administrador.

Cas d'ús 17 "baixa client", classes: administrador.

Cas d'ús 18 "modificar client", classes: client *, administrador.

Cas d'ús 19 "alta denominació", classes: administrador, denominació.

Cas d'ús 20 "consulta denominació", classes: administrador, denominació.

Cas d'ús 21 "baixa denominació", classes: administrador, denominació.

Cas d'ús 22 "modificar denominació", classes: administrador, denominació.

Cas d'ús 23 "alta vi", classes: administrador, vi.

Cas d'ús 24 "consulta vi", classes: administrador, vi.

Cas d'ús 25 "modificar vi", classes: administrador, vi.

Cas d'ús 26 "baixa vi", classes: administrador, vi.

Cas d'ús 27 "consulta comanda", classes: administrador, comanda, client, vi.

Cas d'ús 28 "veure comanda", classes: administrador, comanda, línia_comanda, vi.

Per tant, la llista de classes queda amb les següents entitats:
visitant, client, cistell, vi, denominació, comanda, línia_comanda.

4.1.2 Identificació dels atributs de les classes d'entitat

Classe client: nom (String), cognom1 (String), cognom2 (String), email (String), adreça (String), població (String), província (Província), codi postal (String), username (String), password (String).

Classe cistell: username (Integer), IDvi (Integer), quantitat (Integer), import (Float).

Classe vi: nom (String), collita (Integer), preu (Float), celler(String), tipus (String), varietat (String), categoria (String), descripció (String), denominació (Denominacio).

Classe denominació: nom (String), descripció (String), IDdenominacio (Integer).

Classe comanda: user (client), IDcomanda (Integer), data (Date), import_total (Float).

Classe línia_comanda: comanda (comanda), IDlínea_comanda (Integer), vi (Vi), quantitat (Integer), import_línea (Float).

4.1.3 Diagrama de classes

En el diagrama de classes es descriu l'estructura del sistema mostrant les classes d'entitat, els atributs de cadascuna i les relacions estructurals i d'herència que hi ha entre aquestes. El diagrama de classes es utilitzat per crear el disseny conceptual de la informació que es farà servir al sistema, i els components que s'encarregaran del funcionament.

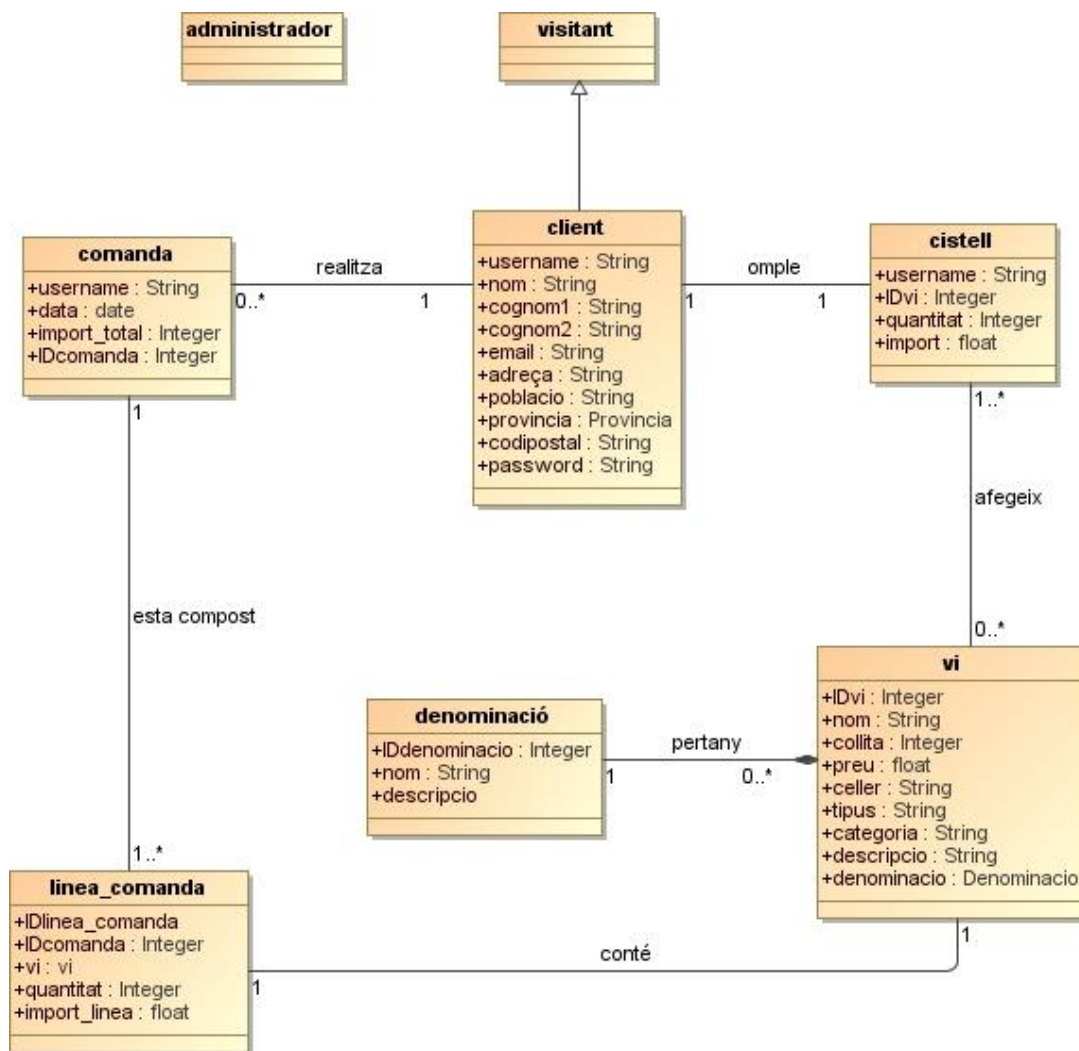
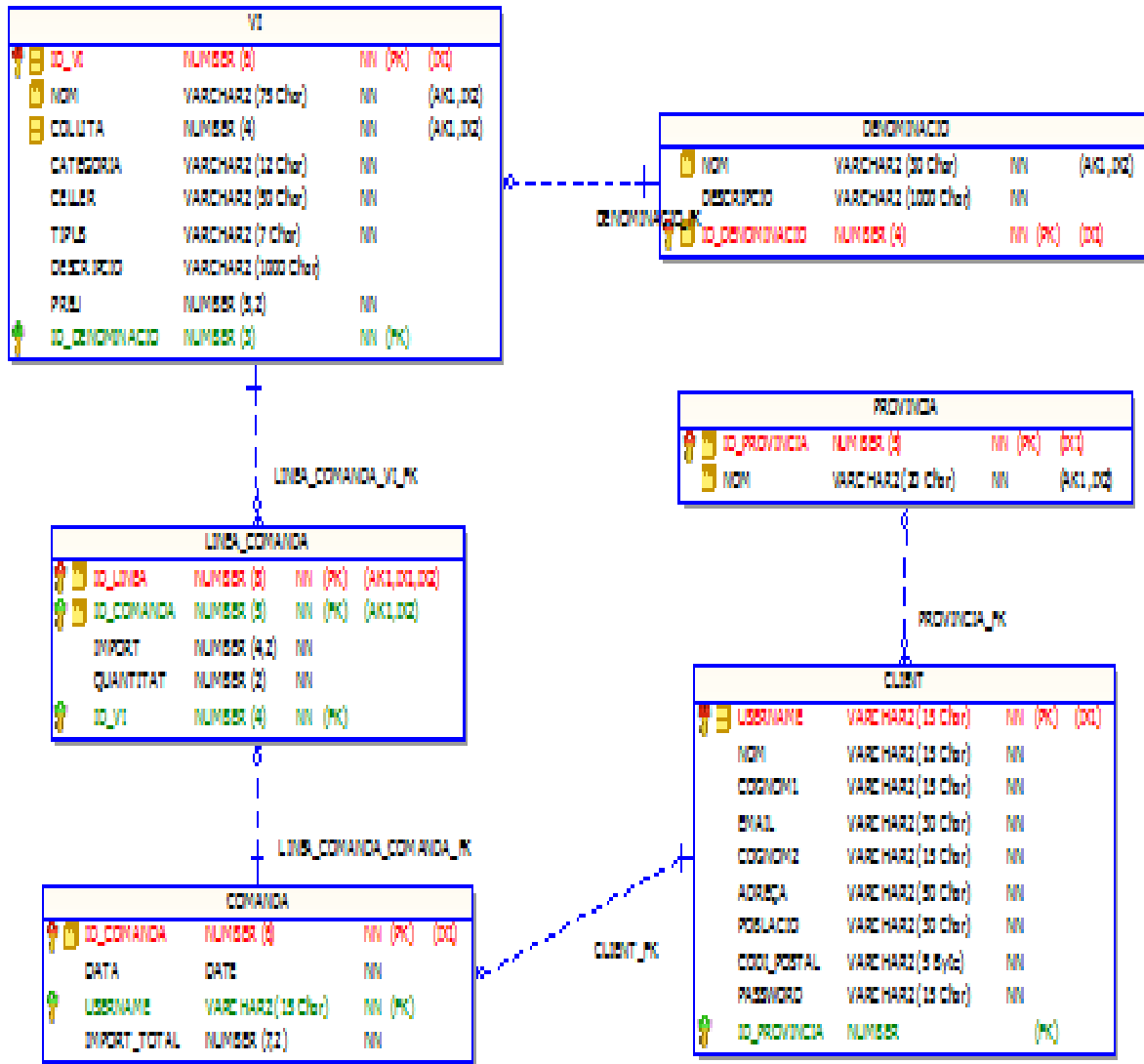


Figura 19. Diagrama de classes.

4.2 Disseny de la base de dades

4.2.1 Disseny relacional

Seguint el model de les bases de dades relacionals, es presenta a continuació el diagrama del disseny de la persistència de dades del projecte. No s'ha tingut en compte l'entitat cistell com a dada persistent, per això no està inclosa dins el model relacional de la base de dades.



Legenda del diagrama:

Claus privades: color vermell amb una clau al costat del nom de l'atribut.

Claus úniques: símbol groc al costat del nom del atribut.

Claus foranies: color verd amb una clau al costat del nom de l'atribut.

Figura 20. Diagrama entitat-relació de la base de dades.

4.3 Arquitectura del projecte

En aquest apartat es presenta la arquitectura definitiva a implementar mitjançant el següent esquema:

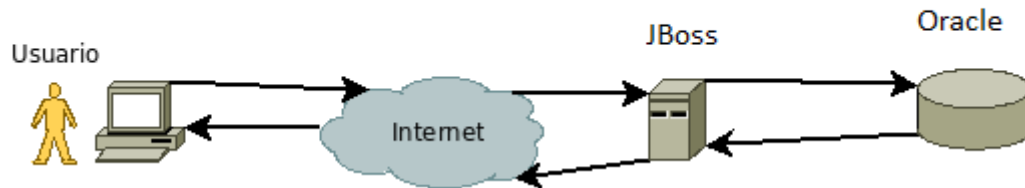


Figura 21. Esquema de l'arquitectura del sistema

Al sistema s'identifiquen els següents elements:

- **Usuari:** representa el client i l'administrador de l'aplicació, que interactuen de la mateixa forma, a través d'un navegador web. Accediran a l'aplicació allotjada en el servidor web i executaran un conjunt d'operacions determinades.
- **Servidor d'aplicacions i contenidor web:** element que allotjarà l'aplicació que implementarà les funcionalitats requerides.
- **Base de dades:** la funció en aquesta arquitectura és la de emmagatzemar les dades que requereixi l'aplicació.

4.3.1 Patró de disseny MVC

El Model Vista Controlador (MVC) és un patró d'arquitectura de software que, per tal de reduir l'acoblament de capes, separa les dades d'una aplicació, l'interfície de l'usuari i la lògica de control en tres components diferents: el primer és un **model** que representa les dades de l'aplicació i les regles de negoci, el segon és un conjunt de **vistes** que representa els formularis per interactuar i el tercer és un **controlador** que processa les peticions dels usuaris i controla el flux d'execució del sistema.

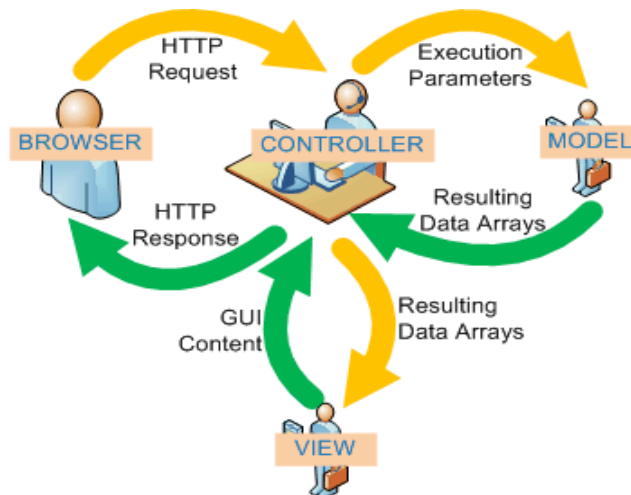


Figura 22. Diagrama de l'arquitectura MVC.

A continuació es presenten els diferents frameworks i tecnologies emprades per implementar el patró MVC:

4.3.2 Struts 2

Actualment Struts 2 és un dels frameworks més emprats per implementar el patró MVC. En primer lloc, hem de remarcar que la versió 2 de Struts està basada en el ja existent framework WebWork, i que el producte final és fruit d'un treball de col·laboració de les dues comunitats: Jakarta i WebWork..

Bàsicament Struts és una eina de suport per al desenvolupament d'aplicacions web fent servir el patró MVC i els avantatges de fer servir aquesta metodologia es la reducció del temps de desenvolupament i el fàcil manteniment de l'aplicació web final:

- Transport automàtic de les dades introduïdes al client (JSP) fins el controlador (Action) mitjançant formularis (ActionForm).
- Transport automàtic de les dades enviades pel controlador (Action) a la part de presentació (JSP) mitjançant formularis (ActionForm).
- Implementa la part comú a totes les aplicacions de la part Controlador (ActionServlet); la part particular de cada aplicació es fàcilment configurable (struts-config.xml).
- La separació dels components en capes (MVC) simplifica notablement el desenvolupament i el seu manteniment.

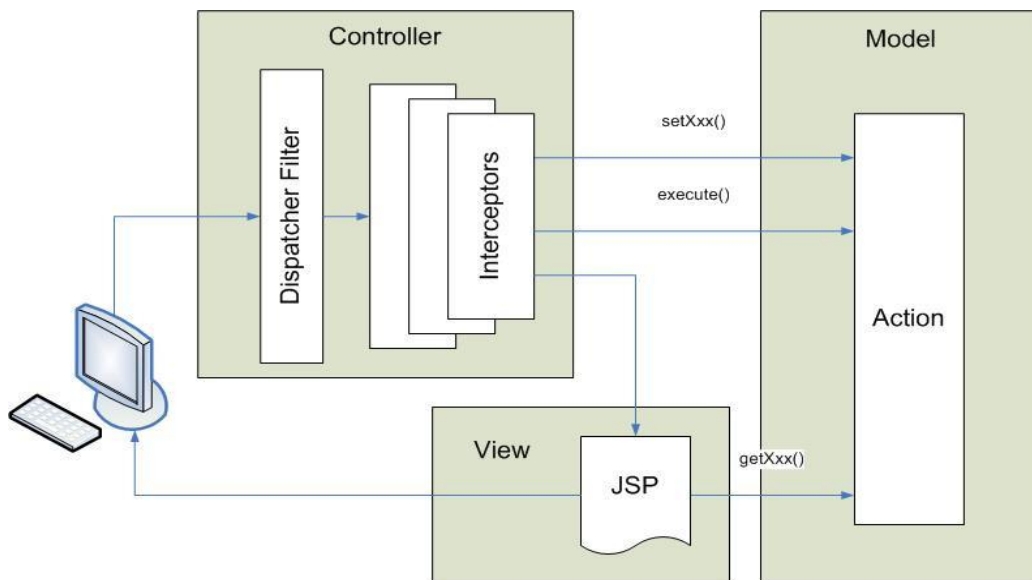


Figura 23. Arquitectura del framework Struts 2.

4.3.2.1 Model

El model de l'aplicació a desenvolupar fa referència a les dades que tracta l'aplicació i les regles de negoci que operen sobre aquests. En Struts 2 són el conjunt d'accions.

Les accions es representen en l'aplicació a través de classes Java. Aquestes es caracteritzen per ser classes simples amb una sèrie d'atributs. Encara que, han de tenir les següents característiques:

- Tenir la funcionalitat que les converteix en accions per funcionar en Struts. Això s'aconsegueix fent que cada classe hereti de la classe abstracta **ActionSupport**, classe proporcionada per Struts 2 que conté el mètode **execute()**, on s'implementa la lògica de negoci de l'acció.
- Ser capaços de treballar amb una sessió d'usuari. Això permet que únicament s'executin les accions en el cas que l'usuari tingui oberta una sessió vàlida amb el sistema. Aquest comportament es adquirint en cada una de les classes (accions) implementant l'interfície **SessionAware**.

El conjunt d'accions a definir en el projecte han de treballar amb el conjunt de classes que engloben el domini de l'aplicació, que són el conjunt d'entitats de dades de l'aplicació (vist al diagrama de la figura 19).

Arribat aquest punt, una bona pràctica de disseny és aplicar el patró Data Access Object (DAO) a les entitats de dades. I consisteix en que el conjunt d'accions del sistema treballin amb classes genèriques i no es preocupin de l'implementació de les mateixes, per si en un futur es pren la decisió de variar la font de les dades però deixant intacta la lògica de negoci.

El DAO actua com un adaptador entre el component i la font de dades, i així es permet un accés transparent, per que els detalls de la implementació s'oculten dins el DAO, també permet migrar amb més facilitat, reduir la complexitat del codi dels objectes de negoci i centralitzar tots els accessos a dades en una capa independent.

Normalment es generen tants DAOs com models d'objectes hi ha, però en aquest projecte s'han agrupat en tres grans blocs: el DAO de clients, el DAO de Productes (referent a les dades de vins i denominacions d'origen) i el DAO de Comandes, on s'agrupen les accions de persistència (afegir, modificar i esborrar) de cada grup de dades. Existeix un DAO de Províncies al projecte, que només tracta el llistat de províncies, però per manca de temps no s'ha pogut afegir al DAO de Clients i fer les proves pertinents.

Fixem-nos en les següents imatges per veure-ho en detall:

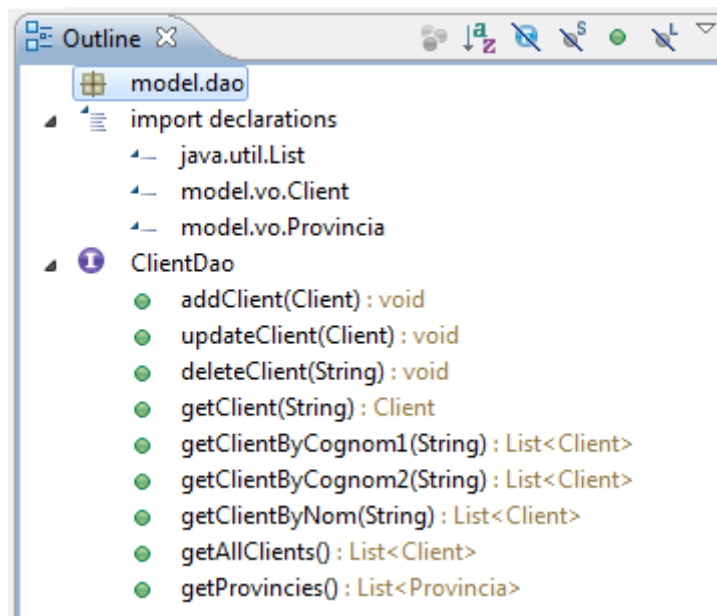


Figura 24. Estructura del DAO de client.

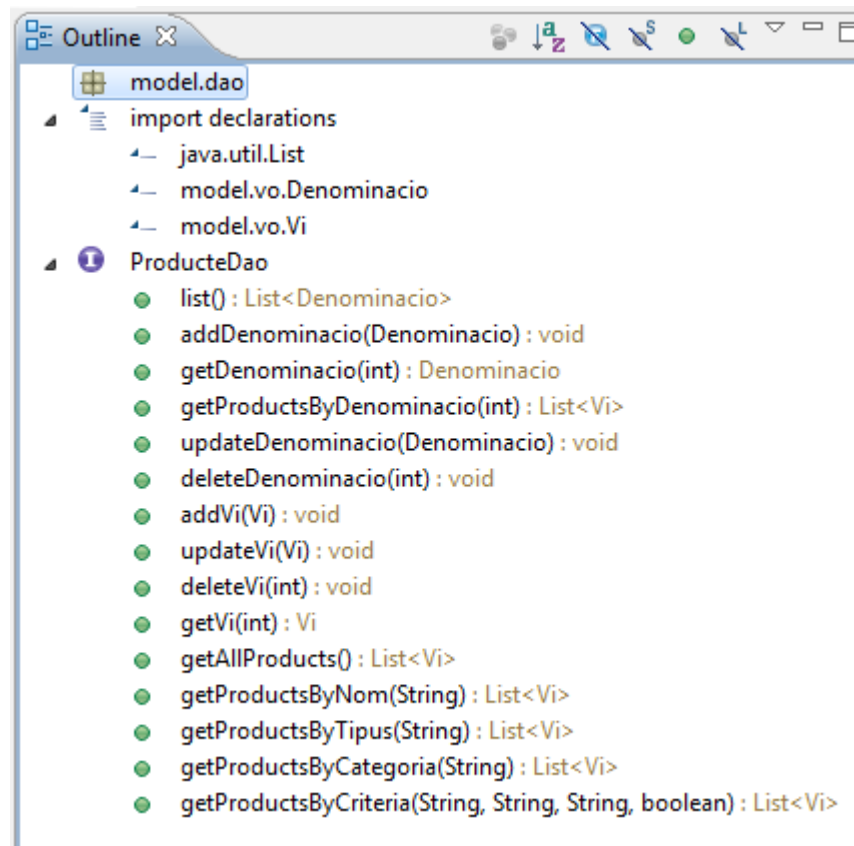


Figura 25. Estructura del DAO de Productes.

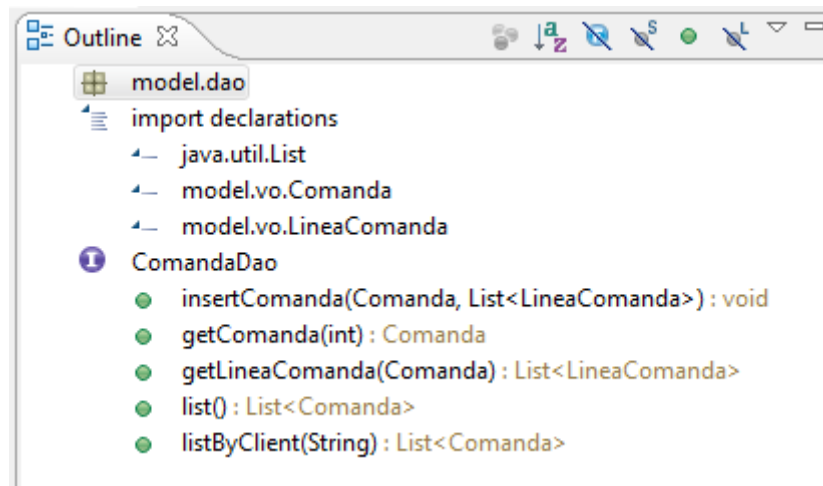


Figura 26. Estructura del DAO de Comandes.

4.3.2.2 Vista

La part de la vista en aquest projecte està organitzat per un conjunt de pàgines JSP, creades per mostrar l'estat del sistema en un determinat moment, tenint en compte que són el medi amb el que l'usuari interactua amb l'aplicació. Per la creació de les pàgines s'han tingut en compte els següents punts:

- Requisits de l'interfície d'usuari ja esmentats a l'apartat 3.4 del present document, i ho aconseguirem mitjançant el framework **Struts Tiles**, de l'Apache Software Foundation, que ens permet tenir parts de la web comunes, com el menú, la capçalera i el peu, i una part de la pàgina dinàmica, que és el cos d'aquesta.
- Fulles d'estil, les pàgines presenten un aspecte similar mitjançant CSS.
- Etiquetes de Struts 2, les pàgines han de fer ús de les etiquetes de Struts 2 per manipular les diferents entitats. Han d'incloure la següent directiva en la capçalera: `<%@ taglib uri="/struts-tags" prefix="s" %>` i d'aquesta forma s'evitarà la introducció de codi Java en la part de presentació.

4.3.2.3 Controlador

El controlador és l'intermediari entre la Vista i el Model, a les peticions de l'usuari a la Vista. Aquest s'encarrega d'invocar al Model qualsevol canvi a les dades de l'aplicació, per que pugui ser retornada a la Vista, com resposta a la petició de l'usuari. En Struts 2, aquesta tasca la realitza el **FilterDispatcher** i conté la lògica de l'aplicació en el fitxer `struts.xml`.

Es pot veure un exemple d'aquest fitxer a la figura 24, i conté el següent:

- El conjunt de possibles crides a l'aplicació web per part de l'usuari.
- Les accions que s'executaran a cada una de les crides.
- Les pàgines que es mostraran en funció del resultat del flux de l'acció a executar.

4.3.3 Tiles

Apache Tiles és un framework construït per facilitar el desenvolupament d'interfície d'usuari d'aplicacions web. Amb una implementació basada en Servlet i JavaServer Pages (JSP), s'identifiquen components HTML i JSP comuns, com la capçalera (Header), el cos (Body), el peu de pàgina (Footer), el menú i la recerca (Search).

Per a estructurar les vistes, es farà servir el sistema de plantilles y esquemes *Tiles*, que es troba disponible com un *plug-in* de Struts 2. Aquest framework utilitza el fitxer de configuració `tiles.xml` per als components de vista (tiles). Els tiles poden ser reutilitzats tantes vegades com es desitgi, aconseguint que el manteniment de les parts comunes en la capa de presentació sigui més senzill i ràpid.

4.3.4 Hibernate

Hibernate és una eina *open source*, que realitza el mapatge entre les aplicacions orientades a objectes i la entitat-relació de les bases de dades en l'entorn Java. Com totes les eines d'aquest tipus, Hibernate intenta solucionar el problema de la diferència entre els dos models de dades coexistents en una aplicació: el de la memòria del computador (orientació a objectes) i el de la base de dades (model relacional). Per aconseguir-ho li permet al programador detallar com és el model de dades, les relacions que existeixen i la forma que tenen.

El paper de Hibernate dins una aplicació Java és aconseguir que el programador s'alliberi de realitzar l'escriptura del codi encarregat de la integració del JDBC i enfoqui la seva atenció en implementar la presentació i la lògica de negocis de l'aplicació.

Hi ha dos requeriments indispensables per al mapatge entre els dos models de dades:

- El primer requeriment són les propietats i es refereix al arxiu de configuració que conté informació sobre quina base de dades es desitja utilitzar, el *driver*⁷ a utilitzar per la comunicació, el port, l'usuari i la contrasenya, és a dir, les dades necessàries per a la connexió amb la base de dades. Aquesta informació es guarda al fitxer `hibernate.cfg.xml`.
- El segon requeriment es el *XML Mapping*, arxius de mapatge entre els *beans*⁸ de Java cap a les taules de base de dades. Aquí s'especifica el tipus de dada que es guardarà, a quina taula es dirigirà aquest *bean*, és a dir, la informació necessària per que Hibernate interactuï amb la base de dades. Aquesta informació es guarda al fitxer `NomClase.hbm.xml`.

4.3.5 DisplayTag

DisplayTag és una llibreria de tags JSP de codi obert que serveix de gran utilitat quan es vol mostrar una col·lecció de dades en una taula, permetent entre d'altres coses, agregar estils, decoradors de les dades mostrades, exportar a diferents formats, utilitzar la paginació i ordenació. Per a consultar els paràmetres de configuració de la llibreria hem d'obrir el fitxer `displaytag.properties`.

4.3.6 Log4j

Log4j forma part del projecte Apache i permet generar logs d'una manera flexible i configurable. Log4j és una biblioteca que permet al programador escollir la sortida i el nivell de l'interval entre missatges en temps d'execució i no en temps de compilació, com es realitza sovint. La configuració del framework es pot consultar i modificar al fitxer `log4j.properties`.

⁷ Software per poder controlar un dispositiu hardware.

⁸ Component software, és reutilitzable i ha de tenir els atributs privats, mètodes `get()` i `set()` públics dels atributs privats i tenir un constructor públic per defecte.

5 Implementació

L'objectiu d'aquesta etapa es desenvolupar l'implementació de l'aplicació assolint els criteris especificats en les fases d'anàlisi i disseny. Es descriuen els aspectes més importants del programari i eines utilitzades per organitzar l'entorn de treball. Per al bon funcionament del sistema, el software ha de complir uns requisits bàsics.

5.1 Eines de desenvolupament

El projecte estarà implementat mitjançant un sistema operatiu de la plataforma **Windows**, la versió **7 Professional**. A continuació es presenta el programari i eines de desenvolupament que seran utilitzades per a cada capa:

- L'entorn de treball o IDE escollit per a desenvolupar el projecte és **Eclipse 3.5 Galileo**. Per la realització dels diagrames UML com són els casos d'ús, el disseny de la base de dades i el diagrama de classes, es faran servir les eines CASE: **MagicDraw** i **Enterprise Architect**.
- **Java Development Kit 1.6.0-22**, que conté la màquina virtual necessària per l'execució del servidor d'aplicacions.
- Els navegadors web provats per accedir a l'aplicació són **Internet Explorer 8** i **Mozilla Firefox 4**. Per a realitzar els prototips navegables de pàgines web es farà servir el programa **Axure RP**.
- S'utilitzarà **JBoss 5.1** com a servidor d'aplicacions. A més de ser el contenidor de EJB, integra també un servidor web, que es **Apache Tomcat**, per tal de ser el contenidor dels JSP i servlets. És considerat el servidor d'aplicacions lliure per excel·lència, i està molt estès al món empresarial.
- Com a sistema de gestió de la base de dades relacional es farà servir el servidor **Oracle 10g**. Aquest és considerat com el sistema més complet destacant la seva estabilitat, l'escalabilitat, el suport transaccional i la multiplataforma. Farem servir la versió *Express Edition*, que és l'única edició gratuïta de la companyia Oracle al mercat.
- L'aplicació de desenvolupament SQL i administració de la base de dades que es fa servir és **Toad for Oracle 10**. Es fa necessari tenir instal·lat el client d'Oracle per a Windows per tenir connexió amb la base de dades Oracle.

5.2 Decisions d'implementació

Durant la fase d'implementació s'han fet algunes modificacions del producte dissenyat originalment degut a problemes que no havien estat contemplats inicialment, millores en l'eficiència o qüestions referents als frameworks escollits per la realització del projecte.

Un exemple en la dificultat d'implementació ha estat l'opció de tenir un llistat de les denominacions en el menú lateral esquerra mitjançant el *tag select* que proporciona Struts 2, per tal de mostrar un llistat dels vins d'aquesta denominació concreta. La solució ha estat generar un botó per llistar les denominacions en una pàgina JSP amb una taula implementada en DisplayTag i dotar a cada denominació d'un vincle al llistat de vins corresponent.

El Look & Feel⁹ d'algunes de les finestres de l'aplicació no estan del tot presentades com es desitjava, l'ús de les validacions de formularis ha obligat a mantenir l'aspecte predeterminat d'aquests formularis. Un exemple d'aquestes finestres són les d'alta o modificació de les dades del sistema. Les validacions dels Action les he realitzat de forma declarativa en Struts 2, seguint la convenció d'un arxiu XML denominat "class_del_action-validation.xml" en el mateix paquet on es troba el Action. Abans de cridar al Action, es realitzen les validacions dels camps, si no passen totes les validacions, no s'executa el Action i es crida al result amb l'atribut name="input" per ubicar la vista a mostrar. En el formulari del JSP, si volem fer servir aquest tipus de validacions, no podem canviar el tema emprat per defecte.

Podem veure un exemple d'un fitxer XML de validacions del projecte en la següent imatge:



```
1<?xml version="1.0" encoding="UTF-8"?>
2<!DOCTYPE validators PUBLIC
3    "-//OpenSymphony Group//XWork Validator 1.0.2//EN"
4    "http://struts.apache.org/dtds/xwork-validator-1.0.2.dtd">
5
6<validators>
7
8    <field name="username">
9        <field-validator type="requiredstring">
10            <message>Has d'introduir el nom d'usuari.</message>
11        </field-validator>
12        <field-validator type="stringlength" short-circuit="true">
13            <param name="maxLength">15</param>
14            <message>Usuari massa llarg.</message>
15        </field-validator>
16    </field>
17
18    <field name="password">
19        <field-validator type="requiredstring">
20            <message>Has d'introduir la contrasenya.</message>
21        </field-validator>
22        <field-validator type="stringlength" short-circuit="true">
23            <param name="maxLength">15</param>
24            <message>Contrasenya massa llarga.</message>
25        </field-validator>
26    </field>
27
28</validators>
```

Figura 27. Detall del fitxer LoginAction-validation.xml.

Un punt de l'implementació que ha estat determinant ha estat la part del cistell de compra, he decidit tenir una classe Cistell que contindrà la col·lecció de línies de venda que va afegint el client, i una altra classe CistellProducte per afegir les línies de venda del cistell de compra: el producte, el preu i l'import. El cistell residirà de forma temporal com una variable de sessió del client autenticat. Una vegada expirada la sessió les dades desapareixen de memòria.

⁹ Definir com es presenta una interfície, com es veu i les sensacions que produeix.

5.3 Estructura del projecte

La estructura de directoris de l'aplicació és la següent:

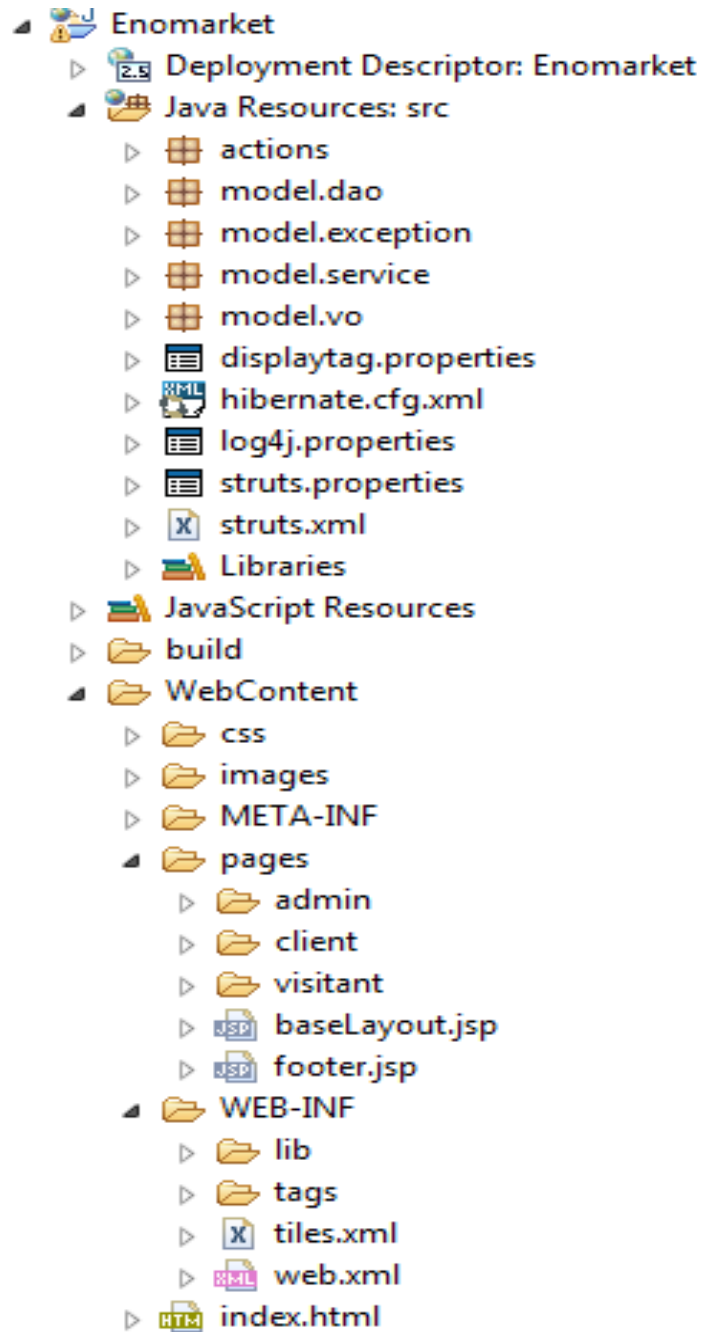
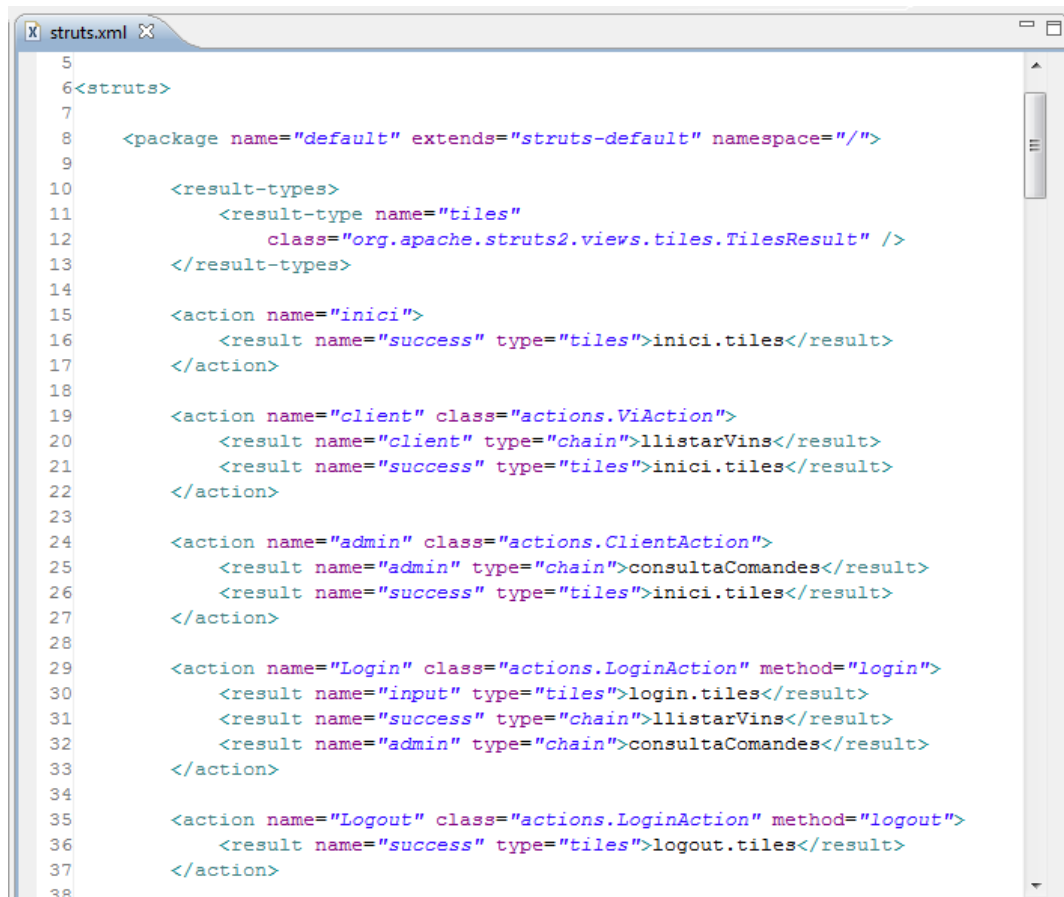


Figura 28. Estructura de directoris del projecte web.

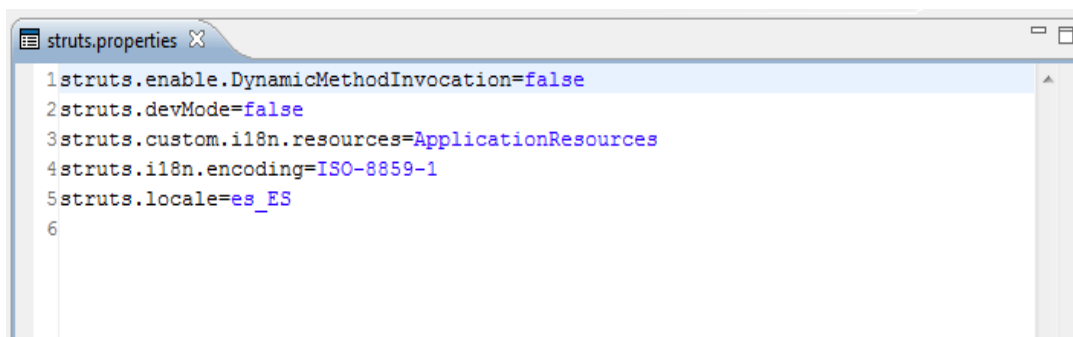
La carpeta "src" conté les classes Java i els recursos de l'aplicació: Actions, POJO, DAO, etcètera. Un dels arxius importants és el fitxer de configuració **struts.xml**, que és utilitzat per iniciar els recursos del projecte, com poden ser els interceptors, actions, results i d'altres.



```
5
6<struts>
7
8  <package name="default" extends="struts-default" namespace="/">
9
10   <result-types>
11     <result-type name="tiles"
12       class="org.apache.struts2.views.tiles.TilesResult" />
13   </result-types>
14
15   <action name="inici">
16     <result name="success" type="tiles">inici.tiles</result>
17   </action>
18
19   <action name="client" class="actions.ViAction">
20     <result name="client" type="chain">l1listarVins</result>
21     <result name="success" type="tiles">inici.tiles</result>
22   </action>
23
24   <action name="admin" class="actions.ClientAction">
25     <result name="admin" type="chain">consultaComandes</result>
26     <result name="success" type="tiles">inici.tiles</result>
27   </action>
28
29   <action name="Login" class="actions.LoginAction" method="login">
30     <result name="input" type="tiles">login.tiles</result>
31     <result name="success" type="chain">l1listarVins</result>
32     <result name="admin" type="chain">consultaComandes</result>
33   </action>
34
35   <action name="Logout" class="actions.LoginAction" method="logout">
36     <result name="success" type="tiles">logout.tiles</result>
37   </action>
38
```

Figura 29. Detall del fitxer struts.xml del projecte.

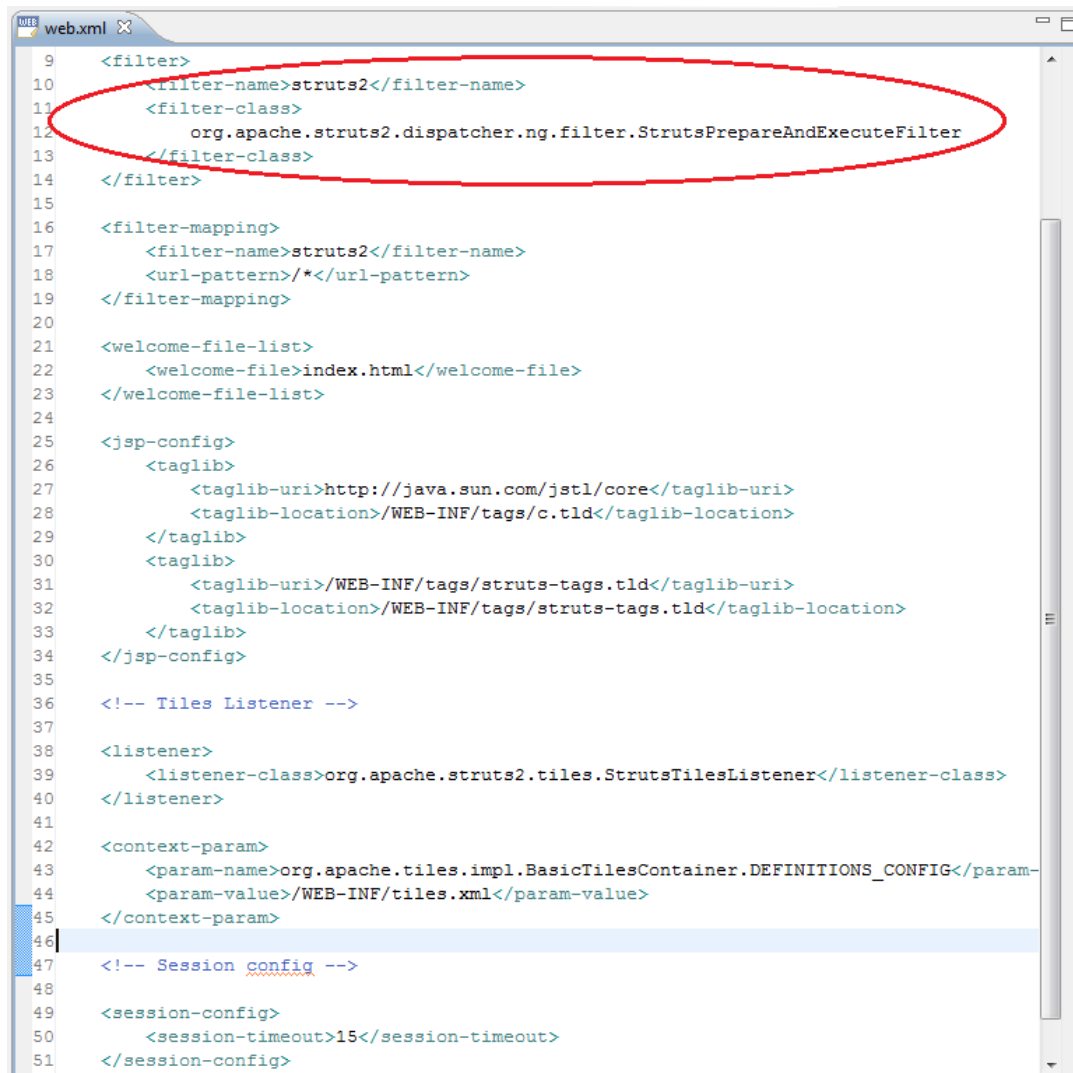
Existeix també un arxiu de propietats, **struts.properties**, que conté un conjunt de valors que permeten definir un conjunt de paràmetres del framework



```
1struts.enable.DynamicMethodInvocation=false
2struts.devMode=false
3struts.custom.i18n.resources=ApplicationResources
4struts.i18n.encoding=ISO-8859-1
5struts.locale=es_ES
6
```

Figura 30. Detall del fitxer struts.properties del projecte.

Un altre dels fitxers de configuració importants del projecte és **web.xml**, que està contingut a la carpeta WebContent/WEB-INF. Aquest és l'encarregat d'indicar al contenidor web quin és el filtre que ha d'executar per a totes les peticions de l'aplicació web, així com la pàgina de benvinguda, l'escoltador de Tiles i més paràmetres del projecte.

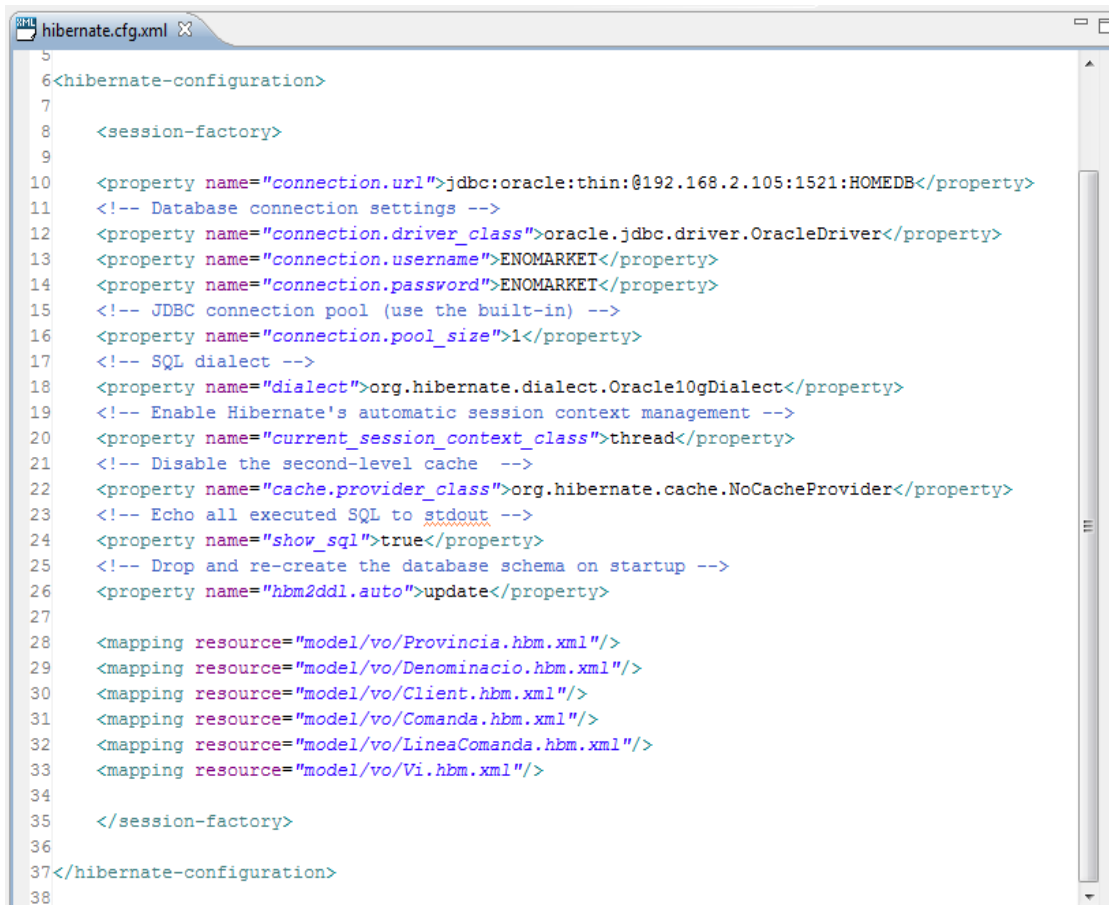


```
9 <filter>
10 <filter-name>struts2</filter-name>
11 <filter-class>
12     org.apache.struts2.dispatcher.ng.filter.StrutsPrepareAndExecuteFilter
13 </filter-class>
14 </filter>
15
16 <filter-mapping>
17 <filter-name>struts2</filter-name>
18 <url-pattern>*</url-pattern>
19 </filter-mapping>
20
21 <welcome-file-list>
22 <welcome-file>index.html</welcome-file>
23 </welcome-file-list>
24
25 <jsp-config>
26 <taglib>
27 <taglib-uri>http://java.sun.com/jstl/core</taglib-uri>
28 <taglib-location>/WEB-INF/tags/c.tld</taglib-location>
29 </taglib>
30 <taglib>
31 <taglib-uri>/WEB-INF/tags/struts-tags.tld</taglib-uri>
32 <taglib-location>/WEB-INF/tags/struts-tags.tld</taglib-location>
33 </taglib>
34 </jsp-config>
35
36 <!-- Tiles Listener -->
37
38 <listener>
39 <listener-class>org.apache.struts2.tiles.StrutsTilesListener</listener-class>
40 </listener>
41
42 <context-param>
43 <param-name>org.apache.tiles.impl.BasicTilesContainer.DEFINITIONS_CONFIG</param-
44 <param-value>/WEB-INF/tiles.xml</param-value>
45 </context-param>
46
47 <!-- Session config -->
48
49 <session-config>
50 <session-timeout>15</session-timeout>
51 </session-config>
```

Figura 31. Detall del fitxer web.xml del projecte.

Els paràmetres de connexió entre el projecte i la base de dades estan configurats al fitxer **hibernate.cfg.xml**. Es configura la connexió JDBC indicant la base de dades, el port, usuari i contrasenya entre d'altres, i s'indiquen els *mapping* existents de classes i taules relacionals.

A continuació veiem un exemple dels dos fitxers de configuració descrits:



```
hibernate.cfg.xml
5
6<hibernate-configuration>
7
8  <session-factory>
9
10 <property name="connection.url">jdbc:oracle:thin:@192.168.2.105:1521:HOMEDB</property>
11 <!-- Database connection settings -->
12 <property name="connection.driver_class">oracle.jdbc.driver.OracleDriver</property>
13 <property name="connection.username">ENOMARKET</property>
14 <property name="connection.password">ENOMARKET</property>
15 <!-- JDBC connection pool (use the built-in) -->
16 <property name="connection.pool_size">1</property>
17 <!-- SQL dialect -->
18 <property name="dialect">org.hibernate.dialect.Oracle10gDialect</property>
19 <!-- Enable Hibernate's automatic session context management -->
20 <property name="current_session_context_class">thread</property>
21 <!-- Disable the second-level cache -->
22 <property name="cache.provider_class">org.hibernate.cache.NoCacheProvider</property>
23 <!-- Echo all executed SQL to stdout -->
24 <property name="show_sql">true</property>
25 <!-- Drop and re-create the database schema on startup -->
26 <property name="hbm2ddl.auto">update</property>
27
28 <mapping resource="model/vo/Provincia.hbm.xml"/>
29 <mapping resource="model/vo/Denominacio.hbm.xml"/>
30 <mapping resource="model/vo/Client.hbm.xml"/>
31 <mapping resource="model/vo/Comanda.hbm.xml"/>
32 <mapping resource="model/vo/LineaComanda.hbm.xml"/>
33 <mapping resource="model/vo/Vi.hbm.xml"/>
34
35 </session-factory>
36
37</hibernate-configuration>
38
```

Figura 32. Detall del fitxer hibernate.cfg.xml del projecte.



```
Client.hbm.xml
1<?xml version="1.0" encoding="UTF-8"?>
2<!DOCTYPE hibernate-mapping PUBLIC "-//Hibernate/Hibernate Mapping DTD 3.0//EN"
3"http://hibernate.sourceforge.net/hibernate-mapping-3.0.dtd">
4
5<hibernate-mapping package="model.vo">
6  <class name="Client" table="CLIENT">
7    <id name="USERNAME" column="USERNAME">
8      <generator class="assigned"/>
9    </id>
10    <property name="PASSWORD"/>
11    <property name="NOM"/>
12    <property name="COGNOM1"/>
13    <property name="COGNOM2"/>
14    <property name="EMAIL"/>
15    <property name="DIREC" column="ADREÇA"/>
16    <property name="POBLACIO"/>
17    <property name="CODI_POSTAL"/>
18    <many-to-one class="model.vo.Provincia" lazy="false"
19      name="PROVINCIA" column="ID_PROVINCIA" />
20  </class>
21</hibernate-mapping>
22
```

Figura 33. Detall del fitxer Client.hbm.xml del projecte.

5.4 Guia d'instal·lació i configuració de l'aplicació

Passem a realitzar la instal·lació del programari necessari per tal de poder desplegar el projecte amb èxit. D'un principi, és necessari instal·lar el Java Development Kit (JDK que inclou el JRE) per poder executar el projecte, disponible en la web: <http://www.oracle.com/technetwork/java/javase/downloads/index.html>.

5.4.1 Entorn de treball

Per al desenvolupament del projecte l'entorn de treball escollit ha estat l'IDE Eclipse. Disponible en la següent adreça web: <http://www.eclipse.org/downloads/packages/eclipse-ide-java-ee-developers/galileo2>, es troba la versió Eclipse Java Enterprise Edition – Galileo.

El marc de treball treballa sobre un sistema operatiu Windows 7 Professional de 64 bits i el navegador suportat és Mozilla Firefox 4, disponible en la web: <http://www.mozilla.com/es-ES/firefox/>

5.4.2 Base de dades

La versió de la base de dades d'Oracle que he fet servir és la 11g (11.1.0.6.0), i és disponible en la adreça web: <http://www.oracle.com/technetwork/database/enterprise-edition/downloads/index.html>.

Una vegada instal·lada la base de dades, es fa entrega dels fitxers de configuració necessaris per tal de fer servir la base de dades tal i com han estat generada al projecte. Aquests s'han d'importar en l'ordre en que són presentats a continuació:

- El fitxer **Tablespace.sql** és l'script que genera el tablespace o espai d'emmagatzematge de la bases de dades d'Oracle. En aquest 'import' s'haurà de modificar a la tercera línia la ruta on es tenen instal·lats els *datafiles* de la base de dades. Al fitxer presentat, al ser un sistema Linux, el directori d'instal·lació és el següent:

```
'/u01/app/oracle/oradata/HOMEDB/ENOMARKET_DATA.dbf'
```

On la part subratllada és el path que s'haurà de modificar per la ruta on estiguin instal·lats o es vulguin instal·lar els *datafiles* d'Oracle.

- El fitxer **Usuari.sql** és l'script que genera l'usuari que es fa servir a Oracle. (S'haurà de modificar la tercera línia IDENTIFIED BY <password> amb el password que es faci servir, al projecte s'ha utilitzat *ENOMARKET*).
- Per últim tenim dues opcions: importar el fitxer **Enomarket.dmp**, que conté les taules, les relacions i les diferents files amb el contingut complet de la base de dades d'Oracle (és la opció més fàcil per tenir al complet l'importació realitzada); o bé importar el fitxer **Enomarket.sql**, que conté només les taules i les relacions de la base de dades. Es recomana importar el dump, per tenir carregada la llista de províncies existents al territori, si no s'haurà de realitzar un import de les dades a la taula PROVINCIA de la base de dades.

5.4.3 Servidor d'aplicacions i contenidor web

El servidor d'aplicacions amb contenidor web Tomcat emprat en el projecte és JBoss versió 5.1.0 GA, disponible en la web: <http://sourceforge.net/projects/jboss/files/JBoss/JBoss-5.1.0.GA/>. Per tal d'executar el servidor hem de configurar les següents variables d'entorn al sistema fent clic al botó dret sobre *Mi PC* > *Propiedades* > *Opciones Avanzadas* > *Variables de entorno*:

Crear la variable JBOSS_HOME amb el següent valor de variable:

```
... \jboss-5.1.0.GA
```

Crear la variable JAVA_HOME amb el següent valor de variable:

```
... \jdk1.6.0_22
```

Crear la variable CLASSPATH amb el valor de variable:

```
... \jdk1.6.0_22\lib\tools.jar
```

Configurar la variable d'entorn PATH afegint la següent ruta:

```
... \jdk1.6.0_22\bin
```

Es fa entrega del fitxer **Enomarket.war** per desplegar el projecte. L'única modificació que s'haurà de realitzar és dins del fitxer **hibernate.cfg.xml**, on s'especifiquen les dades de connexió a la base de dades en Java, així està el fitxer de configuració:

```
<property name="connection.url">jdbc:oracle:thin:@192.168.2.105:1521:HOMEDB</property>  
<property name="connection.driver_class">oracle.jdbc.driver.OracleDriver</property>  
<property name="connection.username">ENOMARKET</property>  
<property name="connection.password">ENOMARKET</property>
```

On s'haurà de modificar el camp vermell amb les dades del servidor que es faci servir, és a dir: el nou servidor, el port i el recurs on és el tablespace de la base de dades. Al camp d'usuari i contrasenya, si hem fet l'importació dels scripts a l'Oracle, no farà falta modificar-ho, en altre cas comprovar les noves dades.

Per a fer un desplegament directe (al no utilitzar recursos de servidor o de correu electrònic) es pot copiar el fitxer **Enomarket.war** al directori on estigui instal·lat el servidor JBoss:

```
... \jboss-5.1.0.GA\server\default\deploy
```

Per iniciar el servidor JBoss, executem el fitxer **run.bat** que és dins el directori:

```
... \jboss-5.1.0.GA\bin
```

Ja estarà tot preparat per començar a treballar amb l'aplicació Enomarket, la direcció web d'inici és:

```
http://localhost:8080/Enomarket/
```

6 Conclusions

Una vegada escollida com àrea de treball per realitzar el Treball de fi de carrera l'arquitectura J2EE, vaig comprovar la quantitat d'especificacions, components de desenvolupament i metodologies que la componen. A diferència d'altres àrees de treball, aquesta requereix més hores de dedicació per tal de portar a terme l'objectiu final: elaboració d'un projecte d'una aplicació web partint de zero.

La corba d'aprenentatge és elevada des d'un principi: es comença decidint les especificacions que es faran servir, la meua experiència amb els frameworks era nul·la abans de començar el projecte, això suposa una dedicació de temps addicional per estudiar les diferents tecnologies que s'implementaran a posteriori, seleccionant les que més ens convinguin.

La valoració personal referent al desenvolupament del projecte, és de molta satisfacció. Al principi, ho veia quasi impossible de poder acabar, el fet d'haver de realitzar la fase d'anàlisi i disseny, l'implementació i la documentació em suposava molt d'esforç, però una vegada realitzades totes les fases, veig que els coneixements de moltes de les assignatures posades en pràctica de manera comú, donen sentit al temps d'estudi realitzat per obtenir un producte acabat. Ha estat essencial seguir les fites i tasques marcades a la planificació per garantir l'entrega en el període delimitat.

És molt enriquidora la tasca personal de realitzar l'estudi dels frameworks més comuns i decidir-ne quin utilitzar en cada cas, fent nosaltres mateixos la valoració del que és convenient. Em vaig decantar per fer ús de Struts 2 com a marc principal de treball, ja que segueix el patró de disseny MVC, i separa el codi HTML del codi Java en les JSP, permetent el manteniment d'aquest amb més facilitat. Per treballar amb la persistència de dades del sistema, Hibernate ha estat una bona opció per ser el framework de codi obert més madur i més complet que hi ha al mercat, amb el suport de la gran comunitat que es troba a la xarxa.

Els requeriments exposats com a objectius del TFC s'han aconseguit finalitzar dins la planificació marcada, encara que per manca de temps no he pogut i m'hagués agradat aprofundir en alguns temes com: la incorporació d'imatges a les fitxes de vins, l'ús dels interceptors de Struts 2 o millorar l'aspecte visual de l'aplicació.

7 Glossari

A continuació es presenta una llista representativa dels termes emprats en la comprensió de l'anàlisi i el disseny del projecte.

Administrador: Usuari del sistema amb privilegis per gestionar i administrar les dades de clients, el catàleg de vins i les comandes realitzades.

Botiga virtual: Espai web on les persones registrades poden realitzar la compra de productes que no es poden veure físicament, fins que la comanda es rebuda pel client.

Catàleg: Són els vins que hi ha a la botiga disponibles per a la seva compra.

Cistell de compra: espai virtual on es guarden els productes que un client vol comprar.

Client: Actor conegut pel sistema (usuari registrat), amb accés a la part pública per poder afegir productes al cistell de compra personal i realitzar comandes.

Comanda: És la sol·licitud de compra dels productes seleccionats al cistell de compra d'un client.

Denominació d'origen: És una entitat que garanteix la qualitat i la certificació de vins d'una mateixa zona.

Framework: És l' implementació de certes solucions comuns, normalment empaquetades, per poder fer més ràpid el desenvolupament.

J2EE: Conjunt d'especificacions que fan possible solucions de desenvolupament, implantació i tractament d'aplicacions multicapa del costat del servidor.

Paginació: Procés de millora de la visualització dels elements a llistar en diverses pàgines, en el cas de tenir un número de resultats extens que és incòmode per a ser visualitzar en una única pàgina.

Visitant: Actor del sistema que no és conegut per l'aplicació i que només pot accedir a la part pública d'aquesta.

8 Bibliografia

- Ian Roughley (2006).** *Starting Struts 2* [llibre en línia].
<http://www.infoq.com/resource/minibooks/starting-struts2/en/pdf/StartingStruts2online2.pdf>
- Donald Brown, Chad Michael Davis, Scott Stanlick (2008).** *Struts 2 in action* [llibre en línia].
http://www.amazon.com/Struts-2-Action-Don-Brown/dp/193398807X/ref=sr_1_1?ie=UTF8&qid=1225291443&sr=8-1#reader_193398807X
- Apache Software Foundation (2011).** *Apache Struts 2 Documentation*.
<http://struts.apache.org/2.0.6/docs/home.html>
- Vaan Nila (2009).** *Struts 2 tutorials*. [en línia].
<http://www.vaannila.com/struts-2/struts-2-tutorial/struts-2-tutorial.html>
- Viral Patel (2009).** *Introduction to Struts 2 Framework*. [en línia].
<http://viralpatel.net/blogs/2009/12/introduction-to-struts-2-framework.html>
- RoseIndia (2008).** *Struts 2 Tutorial* [en línia].
<http://www.roseindia.net/struts/struts2/index.shtml>
- SourceForge.net (2010).** *Display Tag library 1.2*. [en línia].
<http://www.displaytag.org/1.2/>
<http://demo.displaytag.org/displaytag-examples-1.1/index.jsp>
- Cristian Bauer, Gavin King (2007).** *Java Persistence with Hibernate*. Greenwich: Manning.
- Suárez González, Héctor (2003).** *Manual Hibernate*. [en línia].
<http://www.javahispano.org/contenidos/archivo/77/ManualHibernate.pdf>
- JavaScript-Array.com (2006-2010).** *Simple Drop-Down Menu 2.0*. [en línia].
http://javascript-array.com/scripts/simple_drop_down_menu/
- Gavin King, Christian Bauer, Max Rydahl Andersen, Emmanuel Bernard, Steve Ebersole, i Hardy Ferentschik (2004).** *Hibernate – Persistència relacional per Java*. [en línia].
<http://docs.jboss.org/hibernate/core/3.6/reference/es-ES/html/>
- Mkyong (2010).** *Struts 2 + Hibernate Integration example*.
<http://www.mkyong.com/struts2/struts-2-hibernate-integration-example/>

9 Annexos

- Per la realització de proves amb les dades ja existents a la base de dades, són disponibles els següents usuaris:

Accés part pública: USERNAME: emeliac
PASSWORD: emeliac

Accés part privada: USERNAME: admin
PASSWORD: admin