

Universitat
Oberta
de Catalunya

Assignatura:

Treball Final de Carrera

PROJECTE

COMMUSIC

Fase 4: Memòria

Estudiant: Victor Hill Ibar

Titulació: ETIG

Consultor: Josep Juan Rodríguez

Data Lliurament: 18 de Juny de 2007

Universitat Oberta de Catalunya

TFC - AULA 2

Dedicatòria i Agraïments

Al llarg d'aquest Treball Final de Carrera han estat juntament amb mi un bon numero de persones que m'han estat recolzant en tot moment.

He desenvolupat el projecte principalment a casa meva, però també en el meu lloc de treball dins d'Indra, l'empresa que treballo des de fa poc més de 3 mesos. Per aquest motiu, m'agradaria, en primer lloc, agrair a l'empresa el haver-me deixat realitzar part del meu projecte dins de les seves instal·lacions, quan és un projecte que no manté cap relació amb el projecte que faig dins de la mateixa. Han sigut moltes setmanes i molts records que deixo allà on ja s'havia convertit en una rutina el ser l'últim en abandonar la sala.

Gràcies a tots els meus companys de treball i ex treball per l'apoyo que m'heu donat en tot moment i per els moments que m'heu fet passar. No podria deixar escapar aquesta oportunitat sense nombrar, especialment, al meu amic i company Moisés Fernandez Blanco del que he après molt i ha estat ajudant-me i donant-me suport en tot moment de manera desinteressada. Sense ell, estic segur que tot això no ho hagués aconseguit; per a mi sempre serà el meu mestre.

També vull agrair a tota la meva família tot el que han fet per a mi, no ja només durant l'elaboració del projecte sinó durant el transcurs de tota la carrera. Sobretot als meus pares, que sempre han estat ajuntant-me i donant forces perquè fos constant i pogués combinar els estudis amb el treball. Especialment, al meu avi Fèlix, o'pabuelo' com li diem, en tot moment s'ha preocupat per el transcurs dels meus estudis i com anaven evolucionant; des d'aquí l'animo que s'apunti al curs d'informàtica per a gent gran juntament amb el seu amic. A la meva àvia Lola, que desgraciadament ja fa uns quants anys que no està amb nosaltres, però estic segur que des de el cel li agradarà veure que ja em queda menys per aconseguir el títol universitari. De la mateixa manera, agrair l'interés mostrat per part de la meva tieta Sara, el meu tiet Carlos i la meva tia àvia Juli, en que sempre m'han mostrat el seu suport de cara als estudis i especialment en aquest projecte. Al meu germà David, per la seva paciència i comprensió que han demostrat esser immenses.

No podia acabar aquest apartat sense nombrar a la meva novia Júlia i agrair-li sincerament tota la paciència que ha tingut durant els moments en els que ha tingut que suportar les explicacions que feia totes les peripècies del projecte. Júlia, senzillament, moltes gràcies!!

Per finalitzar, agrair al meu consultor del projecte, Josep Juan Rodríguez, l'oportunitat que m'ha donat per aprendre i provar totes aquestes noves tecnologies.

Resum de Treball Final de Carrera

Ens han encarregat construir un portal Web para *COMMusic*, una famosa empresa de venda de música per Internet. Aquest portal web haurà de permetre l'accés a tres tipus d'usuaris: públic en general, socis i empleats de la botiga. Només ens interessarà emmagatzema informació dels dos últims.

El *site* ofereix diferents funcionalitats segons el tipus d'usuari. Per a determinar el tipus d'usuari, aquest s'haurà de connectar mitjançant un nom d'usuari i una contrasenya; mentre no es realitzi la connexió, aquest usuari serà considerat com públic en general.

Els socis que s'hagin connectat tindran accés a totes les possibilitats que ofereix el nostre portal Web. Un soci sense connectar no tindrà accés a les opcions relacionades amb la compra de música on-line ni a la gestió del seu perfil com a soci.

Per a cada 6 € de compra en *COMMusic* s'acumularà 1 punt, que servirà per aconseguir magnífics regals, encara en aquest projecte no es tractarà la gestió de possibles catàlegs de regals, només s'acumularan els punts en cada compra realitzada. Els punts de cada soci podran ser consultats per ell mateix.

Quan es realitzi qualsevol compra, on prèviament s'hagi consultat la disponibilitat de stock, s'actualitzarà la Base de Dades amb la nova comanda. No es contemplarà la garantia d'efectuar la compra ni de realitzar la reposició dels productes. Aquest portal web, es centrarà bàsicament en actualitzar el stock quan es realitzi una venda.

A continuació passo a detallar de manera general les funcionalitats principals de la pàgina Web:

Usuaris Visitants:

- Cerques de Música per Categoria, per Autor i per Preu.
- Accés a la llista *Top 10* de productes del mes.
- Registre de nous Usuaris.

Usuari Socis:

- Comprar productes mantenint un carro de la compra. Opció només disponible per a usuaris prèviament registrats.
- Consultar els punts acumulats per soci registrat.
- Actualitzar les dades personals.

Aquesta aplicació accedirà a una base de dades MySQL on inicialment ja es guarden dades d'alguns socis i d'alguns dels cd's, mp3 o dvds que es podran comprar per Internet.

La implementació de tota l'aplicació, ha estat realitzada en el llenguatge Java. S'ha obtat per seguir una arquitectura J2EE, d'aquesta manera ens ha permès realitzar una aplicació robusta, escalable i altament reutilitzable.

S'ha decidit utilitzar un patró arquitectònic en capes, concretament en 3 capes, corresponents a les capes de presentació, domini i gestió de dades. Les avantatges d'aquesta arquitectura son la reusabilitat, la portabilitat, la canviabilitat i que les dependències només són locals, és a dir, els canvis en una capa només afecten a aquesta.

Paraules clau. Nom de l'àrea de TFC.

La tecnologia que s'utilitzarà en el desenvolupament de tot el projecte és la tecnologia J2EE.

Java Platform, Enterprise Edition o Java EE, és una plataforma de programació per a desenvolupar i executar software de aplicacions en llenguatge de programació Java amb arquitectura de n nivells distribuïda, basant-se àmpliament en components de software modulars que s'executen sobre un servidor de aplicacions.

Java EE, inclou diverses especificacions de API, tals com JDBC, RMI, e-mail, JMS, Serveis Web, XML, etc, y defineix com coordinar-los. Java EE també configura algunes especificacions úniques per a Java EE per a components. Aquestes inclouen Enterprise Java Beans (EJB), servlets, JavaServer Pages i diverses tecnologies de Serveis Web. Tots aquests conceptes, permeten al desenvolupador crear una Aplicació d'empresa que sigui portable entre diverses plataformes i escalable mentre s'integra amb altres tecnologies. Altres avantatges a destacar serien, per exemple, que el servidor de aplicacions pot utilitzar transaccions, seguretat, escalabilitat, gestió de components, etc. D'aquesta manera els desenvolupadors es poden concentrar més en la lògica de negoci dels components i deixar una mica de banda el manteniment a baix nivell.

0 Índex de contingut

DEDICATÒRIA I AGRAÏMENTS	2
RESUM DE TREBALL FINAL DE CARRERA	3
PARAULES CLAU. NOM DE L'ÀREA DE TFC	4
0 ÍNDEX DE CONTINGUT	5
1 INTRODUCCIÓ	7
1.1 Justificació del TFC: punt de partida i aportació	7
1.2 Objectius del TFC	8
1.3 Motivació del TFC	8
1.4 Enfocament i mètode seguit.....	9
1.5 Productes obtinguts	9
1.6 Descripció de la resta de capítols de la memòria.....	10
2 PLANIFICACIÓ DEL PROJECTE	11
2.1 Descomposició del Projecte en Fases	11
2.2 Recursos de programari.....	12
2.3 Calendari del Projecte.....	13
2.4 Temporització i documentació	15
3 ESPECIFICACIÓ I ANÀLISI DELS REQUERIMENTS	15
3.1 Introducció	15
3.2 Composició de les funcionalitats.....	16
3.3 Casos d'ús	16
3.3.1 Definició dels Actors.....	16
3.3.2 Diagrama de Casos d'Ús	17
3.3.3 Casos d'Ús agrupats per Actors.....	18
3.3.4 Descripció dels Casos d'Ús més rellevants.....	19
4 DISSENY	23
4.1 Model Conceptual	23
4.1.1 Diagrama de Classes	23
4.2 Decisions de Disseny	26
4.2.1 Decisions de l'arquitectura del software	26
4.3 Capa de Presentació.....	28
4.4 Capa de Domini	33
4.5 Capa de Gestió de Dades.....	34

4.5.1	Disseny de la Base de Dades	35
4.5.2	Diagrama ER.....	38
4.6	Disseny de les interfícies	39
4.1.1	Estructura pantalles a través de 'tiles'	39
5	IMPLEMENTACIÓ	46
5.1	Introducció	46
5.2	Elecció de tecnologies	46
5.2.1	Sistema Operatiu.....	46
5.2.2	Servidor Web	47
5.2.3	Sistema Gestor de Base de Dades	47
5.2.4	Eines de desenvolupament	47
5.2.5	Llenguatge en el servidor	47
5.3	Disseny de pàgines Web	48
5.4	Plataforma.....	48
5.5	Detalls d'implementació	48
5.5.1	Internacionalització.....	48
5.5.2	Validació de formularis	49
5.5.3	Estructura de directoris	50
5.6	Proves.....	52
6	CONCLUSIONS.....	53
6.1	Objectius aconseguits	53
6.2	Planificació inicial / real	54
6.3	Estudi econòmic.....	55
6.3.1	Cost dels recursos humans.....	56
6.3.2	Cost dels recursos tecnològics.....	56
6.3.3	Cost del software utilitzat	56
6.4	Línies futures	57
6.5	Coneixements adquirits.....	57
7	GLOSSARI.....	58
8	BIBLIOGRAFIA	59
8.1	Llibres	59
8.2	Pàgines web	59
9	ANNEX I: MANUAL D'INSTAL·LACIÓ.....	60
9.1	Configuració de l'aplicació	60
9.2	Configuració de la Base de Dades.....	60

1 Introducció

1.1 Justificació del TFC: punt de partida i aportació

Per a obtenir el títol d'Enginyer Tècnic en Informàtica de Gestió en La universitat Oberta de Catalunya (UOC) l'alumne ha de realitzar a més de les assignatures del pla d'estudis un projecte final de carrera.

En aquest procés, és necessari buscar un projecte de final de carrera per a, posteriorment, treballar en ell. Abans de començar a buscar un projecte, el primer pas que cal fer és decidir l'àmbit sobre el qual es vol treballar per a, posteriorment, concretar el projecte.

En el meu cas, tenia bastant clar sobre quin àmbit volia treballar. Volia poder estudiar i desenvolupar una aplicació web. És a dir, saber com poder gestionar una aplicació web en totes les seves fases. Anteriorment, havia desenvolupat alguna petita aplicació amb *servlets* i *jsp* però sense massa organització. Sempre ha estat un tema que m'ha interessat i tenia davant meu una oportunitat per a descobrir més a fons el desenvolupament d'aquest tipus d'aplicacions.

Una vegada, vaig tenir assignat el tipus de projecte, J2EE i el consultor del projecte, vaig anar a la trobada presencial que es va fer a Bellaterra. Allí, vaig tenir l'oportunitat de conèixer personalment al consultor on vam estar parlant sobre el tema que havia escollit i les tecnologies que utilitzaria per a desenvolupar el projecte. El tema que vaig escollir per a desenvolupar el projecte, va ser el disseny de una pàgina botiga de música virtual on es poguessin efectuar compres a través d'Internet. En un principi, tenia pensat utilitzar *servlets* i *jsps* per el seu desenvolupament, però parlant amb el consultor, en Josep Juan, em va fer una proposta més ambiciosa e interessant però a la vegada més complicada: utilitzar el framework d'*struts* i *hibernate* per a la capa de base de dades.

Després d'aquesta recomanació del consultor, em vaig decantar en seguir aquest camí. En aquest punt, a part de la motivació per aprendre a desenvolupar aplicacions en entorn web, es va afegir utilitzar dos de les tecnologies més innovadores del moment.

D'aquesta manera és com va néixer aquest projecte. Un projecte en el qual, sota el meu punt de vista, em proporcionaria la capacitat d'aplicar els coneixements adquirits prèviament en la carrera i la possibilitat d'adquirir d'altres nous creant una nova aplicació.

1.2 Objectius del TFC

L'objectiu principal del projecte COMUSIC consisteix a oferir un sistema en el qual l'usuari tingui la possibilitat de consultar fàcilment informació referent a publicacions musicals.

A nivell de funcionalitats, l'aplicació ha de proporcionar un servei en qual es puguin donar d'alta, baixar u modificar les dades de usuaris registrats dins de l'aplicació. A més s'ofereix la possibilitat de realitzar cerques sobre els productes registrats dins de la mateixa aplicació.

Un altre dels objectius principals consisteix a permetre a l'usuari *loginarse* dins del sistema i efectuar compres a través de la web. Per a proporcionar aquesta funcionalitat, s'ha de distingir a l'usuari segons el rol que ocupi proporcionant mecanismes d'autenticació i autorització en l'aplicació. Per tant, serà necessari realitzar un gestió d'usuaris on tot usuari no registrat podrà registrar-se com a soci mitjançant un formulari de registració, on es validaran les seves dades.

A nivell de desenvolupament, el principal objectiu resideix a dissenyar e implementar l'aplicació de la manera més estructurada possible permetent que, les posteriors actualitzacions (afegir noves funcionalitats, modificar les existents, ...) que es puguin realitzar en un futur generin el mínim treball possible. Per això, s'ha intentat seguir en tot moment els estàndards del W3C¹ i desenvolupar el projecte sobre la base de patrons de disseny.

¹ El *World Wide Web Consortium* (W3C) és una organització, creada per Tim Berners-Llig, que produeix estàndards web.

1.3 Motivació del TFC

Són varis els motius que m'han dut a desenvolupar el Treball Final de Carrera i, en concret, la botiga Virtual online *COMUSIC*. En la següent llista es troben resumits:

- Poder escollir un Treball Final de Carrera al meu gust i que em permeti portar a la realitat els conceptes teòrics estudiats durant tot els transkurs de la carrera.
- Treballar sobre un projecte basar en una aplicació web.
- Aprendre noves tecnologies, eines i patrons de desenvolupament.
- Afrontar els problemes que comporta treure endavant un projecte i els inconvenients que van sortint en el dia a dia.
- Tenir la capacitat d'escollir la millor elecció en cada cas.

- Planificar el projecte de manera que es pugui finalitzar en el temps establert.
- Marcar un ritme de treball diari.
- Sincerament, un dels motius és el de poder finalitzar els meus estudis i obtenir el títol d'enginyer tècnic.
- Deixar el projecte estructurat de tal manera que pugui venir una segona persona i augmentar les funcionalitats del mateix projecte el més fàcil possible.

1.4 Enfocament i mètode seguit

L'enfocament i mètode seguit en el transcurs del projecte s'ha basat en el sistema estàndard del cicle de vida del programari, és a dir s'ha anat seguint les següents fases: fase de planificació i especificacions, fase d'anàlisi i disseny de l'arquitectura, fase d'implementació i fase de memòria final.

Dintre de cadascuna d'aquestes fases, s'ha anat avançant pel que fa l'anàlisi, disseny e implementació de l'aplicació, però a la vegada s'ha anat estudiant i adquirint nous coneixements sobre diversos patrons o eines de treball com ara *struts* i *hibernate*.

Aquesta ha sigut la principal dificultat alhora d'implementar l'aplicació, la desconeixença prèvia que es tenia sobre aquests conceptes. Com a conseqüència, l'aplicació final ha tingut diverses modificacions respecte original que es va tenir en compte en la fase de especificacions.

A pesar, de les alteracions mencionades, considero que durant el desenvolupament del projecte, s'ha adquirit uns bons coneixements per a tenir una visió general sobre el patró Model Vista Controlador (MVC), el framework de *Struts* i la capa d'*Hibernate*.

1.5 Productes obtinguts

Al final del desenvolupament del projecte, s'ha obtingut una aplicació web programa amb J2ee, la qual es correspon amb la botiga de música virtual. Juntament amb aquesta aplicació web, hi ha documentació, presentació i informació vinculada amb la base de dades. A continuació passo a detallar-la:

- *Comusic.war*: aquest fitxer conté tot el codi font juntament amb tota l'estructura de directoris referents a la implementació de l'aplicació.
- *Hill_Ibar_Victor_Memoria.pdf*: correspon a la memòria final del projecte, es correspon amb el contingut d'aquest document. Dins de la mateixa hi ha un manual d'instal·lació de l'aplicació.

- *BD_COMUSIC_MySQL.rar*: conté dos parts. L'arxiu *script_Comusic.sql* que correspon a l'arxiu de creació de la base de dades. També inclou una carpeta anomenada COMUSIC, que hi ha els arxius de base de dades per omplir les diferents taules amb informació.
- *Presentació del TFC*: document complementari a la memòria on es destaquen els punts més importants.

1.6 **Descripció de la resta de capítols de la memòria**

A continuació es comentaran conceptes els quals s'han pensat, valorat i argumentat en les fases anteriors; les fases d'anàlisi, disseny e implementació. Aquestes fases anteriors es desglossen esquemàticament per els següents conceptes:

- **Capítol de planificació del projecte:** En aquest capítol , el primer que es fa, és descomposar el projecte en fases. A continuació, es detallen els recursos de programari utilitzats. Es mostra la planificació prevista per a realitzar el projecte. Dins de la mateixa, es pot observar les diferents fases seguides amb els dies establerts per a cada cas. També s'inclou un diagrama de Gantt amb la planificació esmentada.
- **Capítol d'anàlisi de requeriments:** Aquest capítol està dedicat a analitzar, i compondre totes les funcionalitats de l'aplicació. Dins d'aquest capítol, apareixen totes les funcionalitats textuais a partir de les necessitats establertes. A més a més, inclou un apartat amb els casos d'ús, casos d'ús desglossats per actors i els diagrames dels més rellevants.
- **Capítol de disseny:** Capítol dedicat al disseny de l'aplicació. Després d'haver definit les funcionalitats de l'aplicació, s'avança a la fase de detallar com ha de funcionar cadascuna. En aquest capítol, a banda del diagrama de classes, inclou les decisions de disseny i d'arquitectura preses, el disseny del patró MVC detallant les capes de presentació, domini i sistema de base de dades. En els últims apartats, apareix el disseny de la base de dades i el disseny ER. Per acabar tenim el disseny de les interfícies de l'aplicació.
- **Capítol d'implementació:** En aquest capítol es detalla la elecció de tecnologies. Tracta en detall les decisions establertes en la fase de disseny. Concretament es centra en com aplicar el patró de disseny MVC en el desenvolupament del producte. També mostra el detall d'implementació de les pàgines Web. S'explica com es tracta el framework de Struts. Al final del capítol, hi ha un apartat per a les proves realitzades.
- **Capítol de conclusions:** Aquest capítol, centra l'atenció en els objectius aconseguits al llarg de tot el desenvolupament. També es mostra la

planificació inicial i real i quina desviació hi hagut. Inclou un estudi econòmic de tota l'aplicació. Es comenten possibles línies futures per a ampliar la funcionalitat aconseguida fins ara.

- **Capítol glossari:** Aquest capítol, descriu les paraules clau sobre l'aplicació desenvolupada.
- **Capítol bibliografia:** Conté un llistat de llibres i d'enllaços d'Internet que s'han utilitzat per adquirir coneixement dels conceptes i tecnologies que hem hagut d'utilitzar al llarg del desenvolupament del projecte.

A més d'aquests capítols que defineixen la memòria, es pot trobar un annex, on es detalla el manual d'instal·lació de l'aplicació.

2 Planificació del Projecte

2.1 Descomposició del Projecte en Fases

Per a realitzar l'aplicació, ens basarem en la metodologia del cicle de vida del programari, on es poden desglossar 4 fases clarament diferenciades:

- **Fase 1. Planificació i Especificacions.**
 - Elaboració de la planificació amb fites i temporalització.
 - Descripció del TFC.
 - Objectius generals i específics.
- **Fase 2. Anàlisi i Disseny arquitectura TFC.**
 - Recopilació d'informació.
 - Diagrames de casos UML.
 - Diagrames textuais i de col·laboració UML.
 - Diagrama de classes.
 - Esquema navegació Web per l'aplicació.
 - Disseny de les interfícies gràfiques de l'aplicació.
 - Disseny de la base de dades.
- **Fase 3. Implementació.**
 - Implementació de les classes d'entitat.
 - Implementació de les classes DAO i *Bussines Objects*.
 - Implementació de d'interfície gràfica HTML i JSP.
 - Implementació dels Action i ActionForm de Struts.
 - Control d'errors i excepcions.
 - Depuració del codi.
- **Fase 4. Memòria final del TFC.**
 - Conclusions.
 - Elaboració memòria final i entrega.
 - Elaboració presentació.









2.2 Recursos de programari

Durant el transcurs de la realització d'aquest projecte s'utilitzaran els següents recursos, dels quals en detallarem la seva funcionalitat:

- **Eclipse 3.2.0:** IDE Java, per el desenvolupament de tota la programació amb llenguatge Java.
S'utilitzarà el mòdul *WebTools* integrat en el mateix IDE per tota la programació Java per entorn web. A més s'ha utilitzat per afegir la funcionalitat pròpia de *Struts* i *Hibernate*.
- **Apache Tomcat:** serà el servidor Web amb suport de *jspx*.
- **MySQL Server 5.0:** base de dades del projecte.
- **Microsoft Project:** programa utilitzat per a fer les planificacions detallades mitjançant un diagrama de Gantt.
- **Magic Draw:** programa per a realitzar els diversos diagrames UML necessaris, com ara el diagrama de classes, de seqüència, col·laboració, etc.
- **Microsoft Visio:** programa per a realitzar qualsevol esquema o diagrames auxiliars, com ara l'esquema de la Base de Dades.

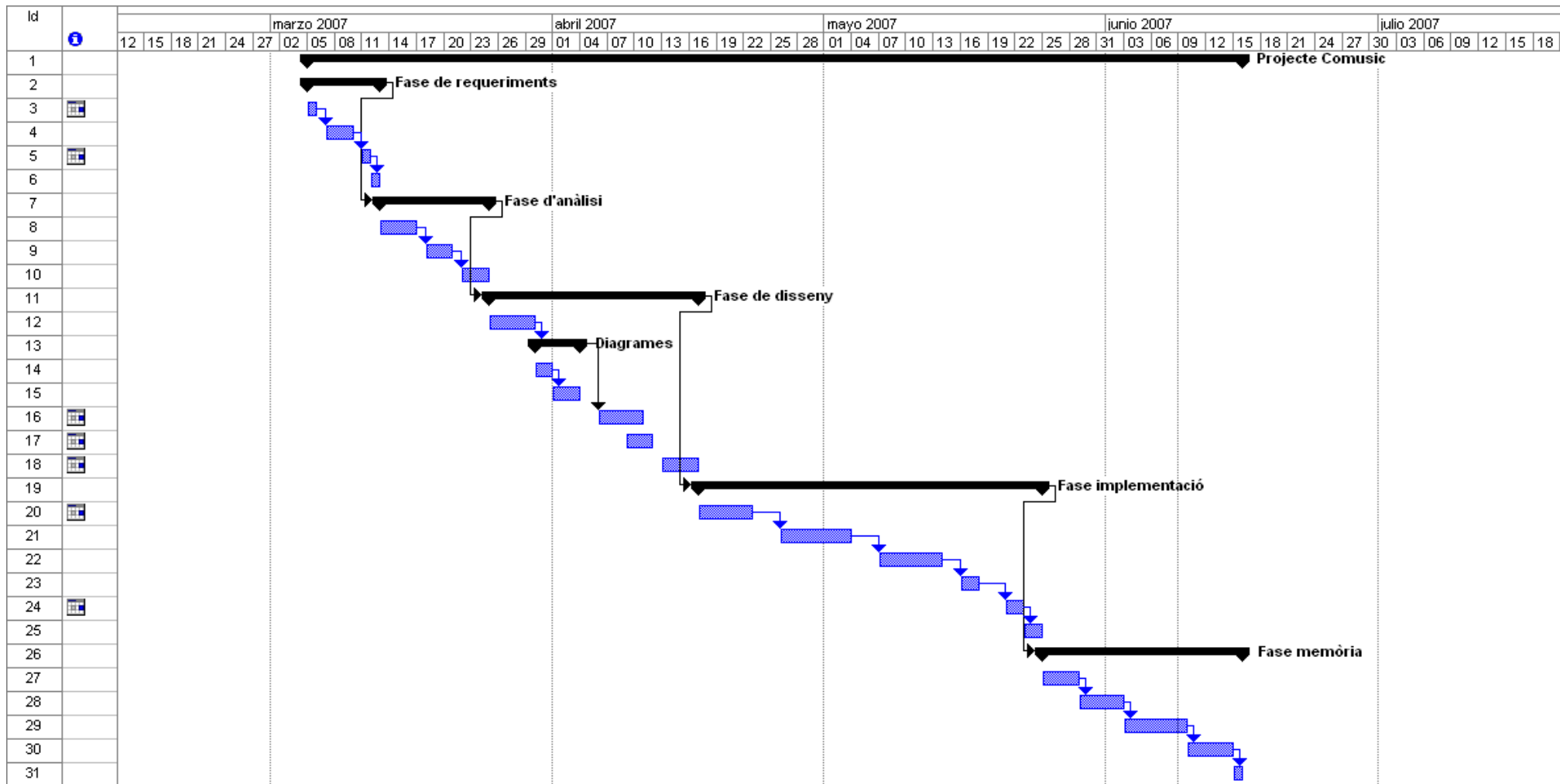
2.3 Calendari del Projecte

A continuació es mostra una captura de pantalla del calendari realitzat amb el programa Microsoft Project. En aquesta imatge, es mostren els períodes de temps que s'han dedicat a cadascuna de les tasques necessàries per a la realització del projecte.

		Nombre de tarea	Duración	Comienzo	Fin	Predecesoras
1		Projecte Comusic	103 días	lun 05/03/07	vie 15/06/07	
2		Fase de requeriments	8 días	lun 05/03/07	lun 12/03/07	
3		Escollir tema del TFC	1 día	lun 05/03/07	lun 05/03/07	
4		Descripció de les necessitats	3 días	mié 07/03/07	vie 09/03/07	3FC+1 día
5		Elaboració del calendari	1 día	dom 11/03/07	dom 11/03/07	4
6		Especificació del projecte	1 día	lun 12/03/07	lun 12/03/07	5
7		Fase d'anàlisi	12 días	mar 13/03/07	sáb 24/03/07	2
8		Especificació funcionalitats	4 días	mar 13/03/07	vie 16/03/07	
9		Casos d'ús	3 días	dom 18/03/07	mar 20/03/07	8FC+1 día
10		Estructura de classes i entitats	3 días	jue 22/03/07	sáb 24/03/07	9FC+1 día
11		Fase de disseny	23 días	dom 25/03/07	lun 16/04/07	7
12		Disseny de les funcionalitats	5 días	dom 25/03/07	jue 29/03/07	
13		Diagrames	5 días	vie 30/03/07	mar 03/04/07	12
14		Classes	2 días	vie 30/03/07	sáb 31/03/07	
15		Casos d'ús	3 días	dom 01/04/07	mar 03/04/07	14
16		Disseny arquitectura	5 días	vie 06/04/07	mar 10/04/07	13FC+2 días
17		Disseny interfície gràfica	3 días	lun 09/04/07	mié 11/04/07	
18		Disseny de la Base de Dades	4 días	vie 13/04/07	lun 16/04/07	
19		Fase implementació	38 días	mar 17/04/07	jue 24/05/07	11
20		Formació prèvia en J2ee, Struts i Hibernate	6 días	mar 17/04/07	dom 22/04/07	
21		Subsistema Visitant	8 días	jue 26/04/07	jue 03/05/07	20FC+3 días
22		Subsistema Soci	7 días	lun 07/05/07	dom 13/05/07	21FC+3 días
23		Depuració Codi	2 días	mié 16/05/07	jue 17/05/07	22FC+2 días
24		Proves i Testing	2 días	lun 21/05/07	mar 22/05/07	23FC+2 días
25		Elaboració documentació i manual	2 días	mié 23/05/07	jue 24/05/07	24
26		Fase memòria	22 días	vie 25/05/07	vie 15/06/07	19
27		Analitzar canvis respecte aplicació original	4 días	vie 25/05/07	lun 28/05/07	
28		Rectificar i corregir modificacions	5 días	mar 29/05/07	sáb 02/06/07	27
29		Elaboració memòria final	7 días	dom 03/06/07	sáb 09/06/07	28
30		Elaboració presentació	5 días	dom 10/06/07	jue 14/06/07	29
31		Entrega final TFC	1 día	vie 15/06/07	vie 15/06/07	30

Imatge 1: Calendari del projecte

A continuació es mostra la mateixa planificació, però amb el diagrama de Gantt:



Imatge 2: Gantt del projecte

2.4 Temporització i documentació

Els períodes de temps que es mostren en el diagrama de Gantt per cadascuna de les fases principals es corresponen amb els períodes que hi ha acordats en el calendari de l'assignatura per els lliuraments de les PACS corresponents. Per tant, podem veure que hi hagut 4 entregues, que es corresponen exactament amb la fase de planificació i especificacions, fase d'anàlisi i disseny, fase d'implementació i fase de memòria.

L'última entrega, la de la memòria final, consta de la documentació del mateix projecte, de l'aplicació desenvolupada i de una presentació. Aquesta entrega, es correspon amb el dilluns dia 18 de Juny de 2007.

A continuació es mostra una taula resumida amb les fases i dates d'entrega fixades:

Fases	Dates de lliurament
Fase de requeriments	12/03/2007
Fase d'anàlisi i disseny	16/04/2007
Fase d'implementació	21/05/2007
Fase de memòria	18/06/2007

3 Especificació i anàlisi dels Requeriments

3.1 Introducció

Aquesta fase, es centra principalment en l'especificació i anàlisi dels requeriments del projecte. Aquesta, és una etapa del procés de enginyeria del software destinades a definir de una manera clara i el més precisa possible les funcionalitats del sistema que es vol desenvolupar evitant possibles problemes que puguin sorgir en un futur, en les etapes de disseny e implementació.

Una vegada definides les necessitats de l'usuari mitjançant l'anàlisi de requisits, és necessari concretar que ha de fer el sistema per a definir aquestes necessitats. L'especificació juga un paper important en aquest sentit.

Per a realitzar l'especificació del projecte s'ha utilitzat la notació *Unified Model Language* (UML). A dia d'avui, UML està considerada la notació estàndard per a modelar els elements d'un sistema programari.

3.2 Composició de les funcionalitats

El projecte exposat es centra en la compra i gestió de cd's de música a través d'Internet. Per tant, aquesta aplicació podrà atendre peticions des de dos punts de vista diferenciats; els Usuaris Visitants i els Usuaris Registrats. La principal diferència entre ambdós usuaris és que el primer només podrà utilitzar la pàgina de mode consulta; en canvi el segon a més a més de poder consultar informació, podrà accedir a una àrea restringida per a usuaris on podrà efectuar comandes i consultar i/o modificar les seves dades personals. D'aquesta manera, l'aplicació web es pot desglossar en dos parts diferenciades:

- **Consulta Botiga Virtual:** en aquesta part, hi podrà accedir tot tipus d'usuari i només serà de mode consulta. Es podrà visitar la pàgina i accedir a tota l'informació pública per a tot tipus d'usuaris.
- **Accés Usuaris Registrats:** en aquesta part, es necessitarà estar prèviament registrat a l'aplicació per accedir-hi; només hi podran accedir usuaris registrats. Dintre d'aquesta secció, podem diferenciar dos tipus d'usuaris:
 - Socis Registrats: aquests tipus d'usuaris, una vegada hagin accedit satisfactòriament dins el sistema mitjançant un nom d'usuari i contrasenya, podran efectuar comandes i modificar les seves dades personals.

3.3 Casos d'ús

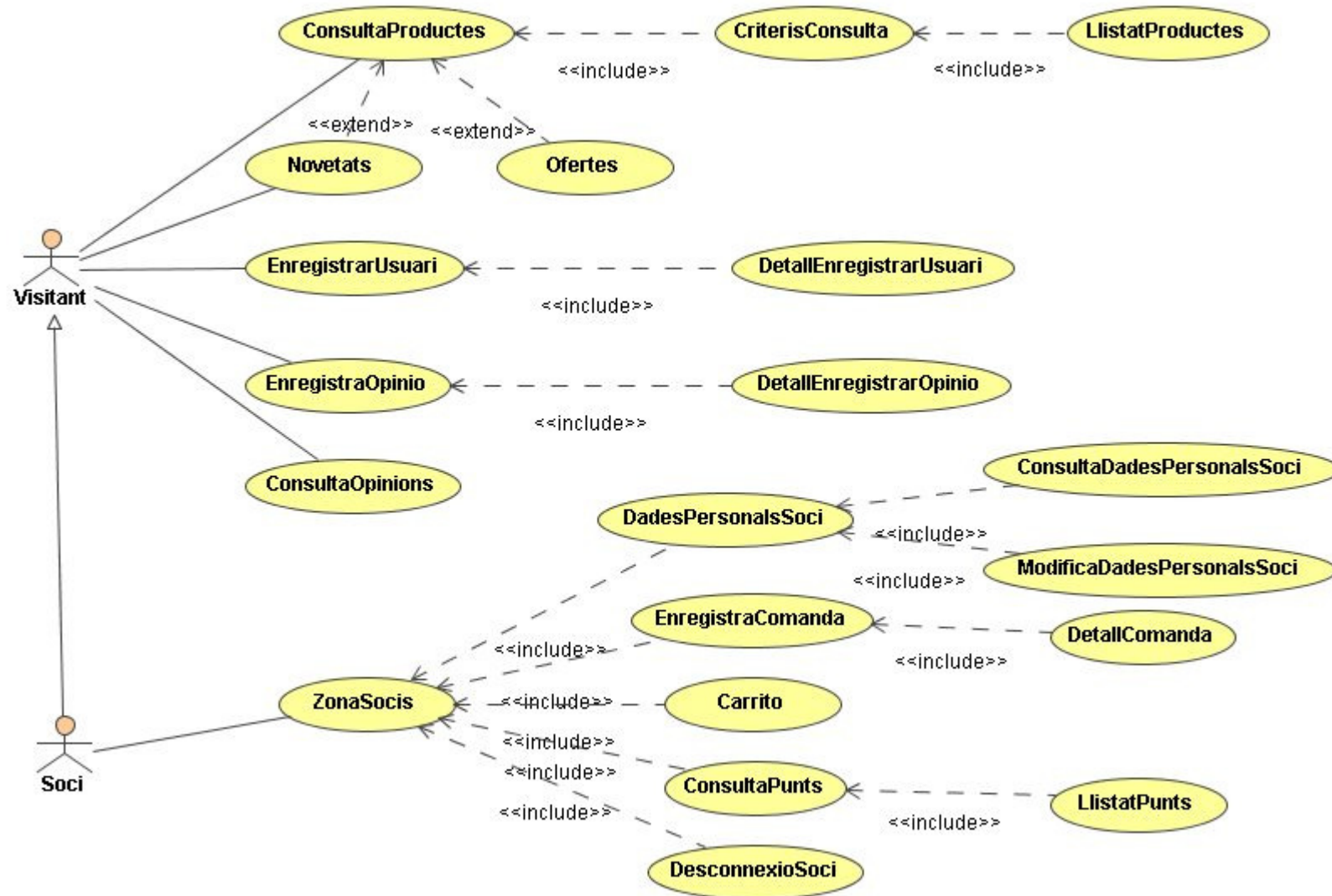
3.3.1 Definició dels Actors

Un actor és una entitat externa al sistema que participa en la història del Cas d'Ús. Aquest pot ser una persona, un conjunt de persones, un sistema hardware, un sistema software, etc.

En el cas de la nostra aplicació Web, es poden distingir els següents actors:

- Visitant: persona que accedeix a l'aplicació i es mou per ella sense registrar-se ni autenticar-se en cap moment.
- Soci: persona que està prèviament registrada al sistema. Aquest tipus d'usuaris, podran accedir a una part de l'aplicació privada on podran efectuar compres.

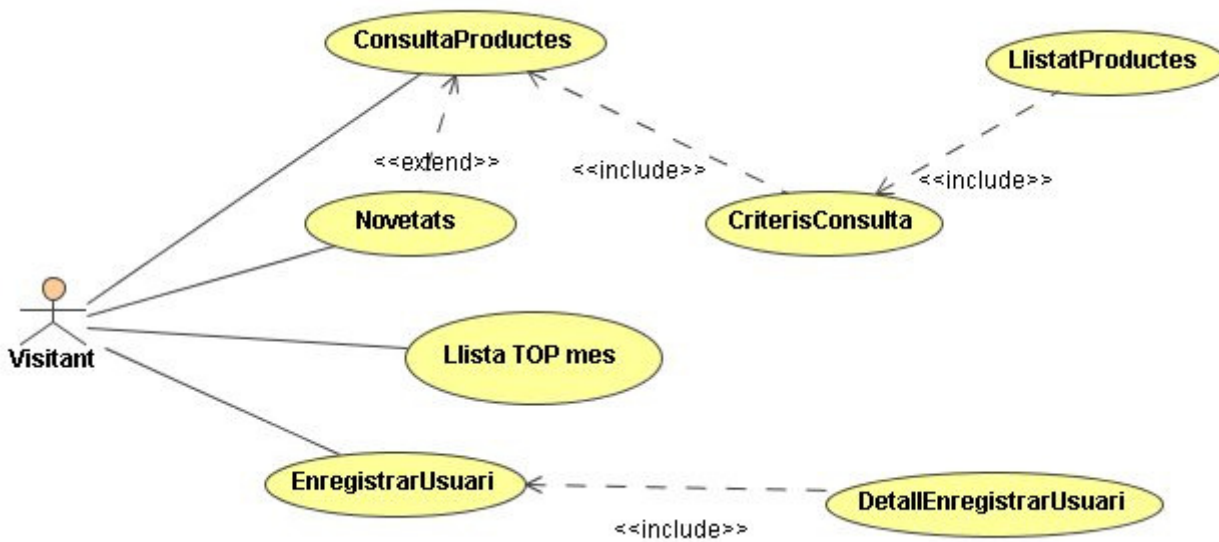
3.3.2 Diagrama de Casos d'Ús



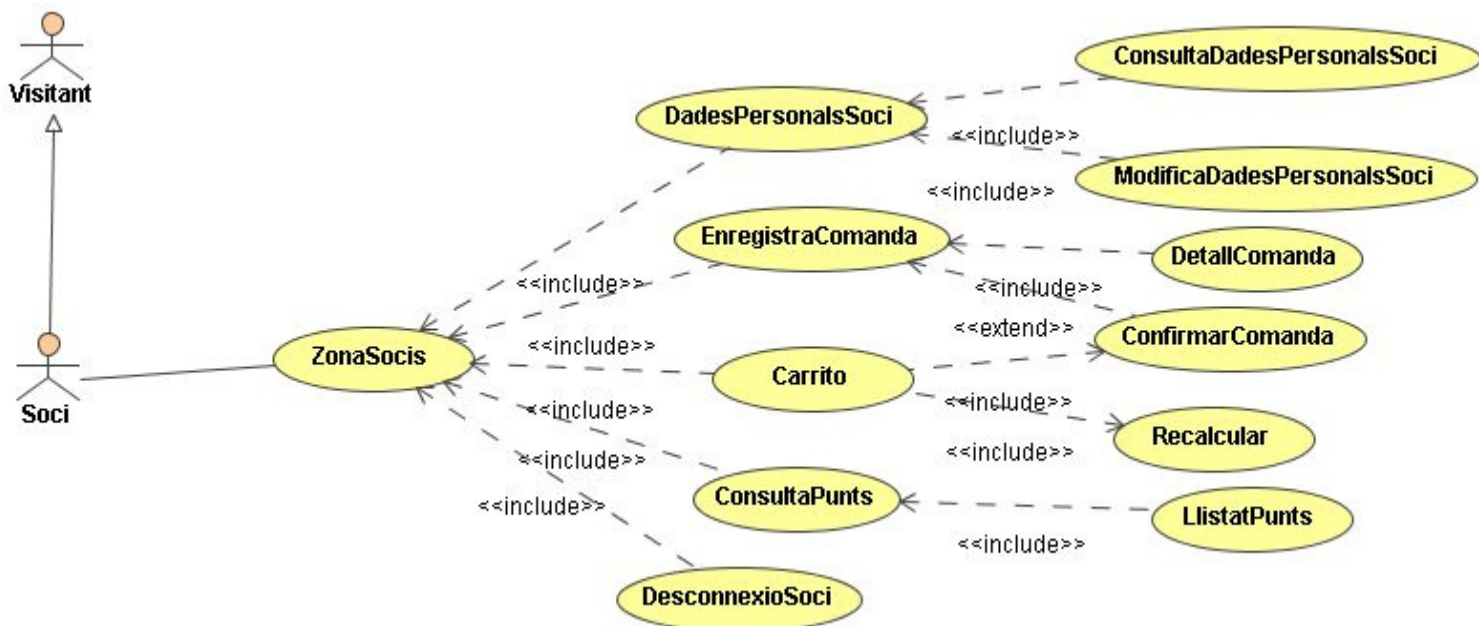
3.3.3 Casos d'Ús agrupats per Actors

A continuació es mostren els casos d'ús generals agrupats per els tres diferents actors.

3.3.3.1 Actor Visitant

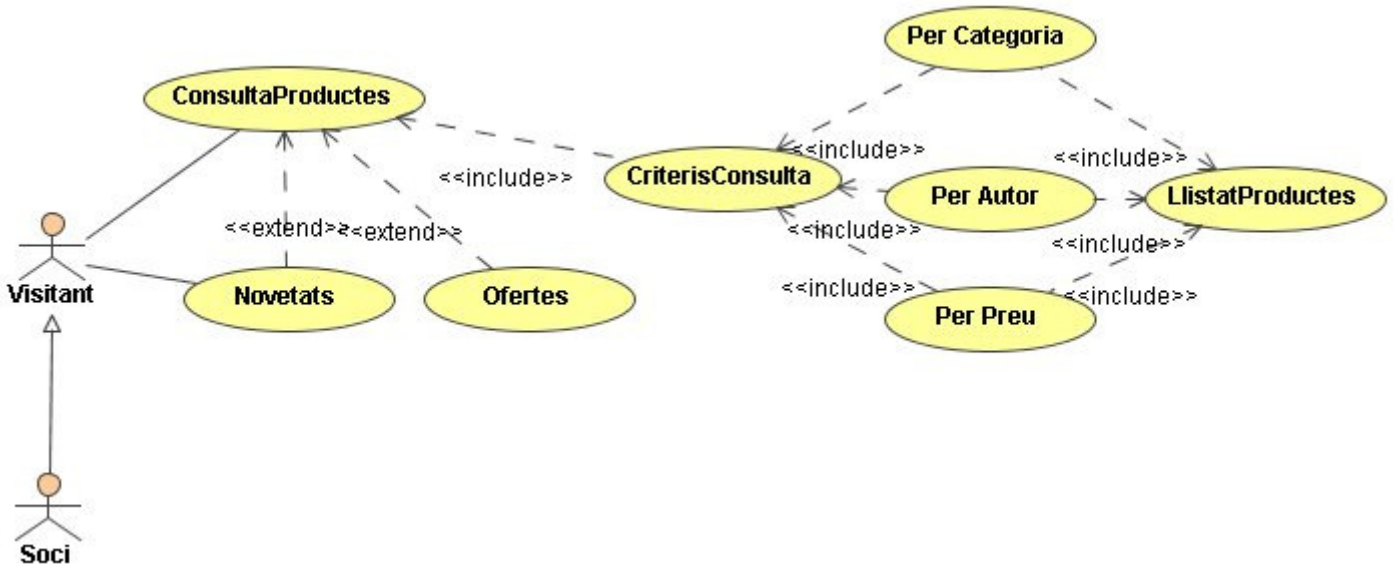


3.3.3.2 Actor Soci



3.3.4 Descripció dels Casos d'Ús més rellevants

3.3.4.1 Cas d'Ús ConsultarProducte



Funcionalitat: Aquesta opció permet als usuaris de qualsevol tipus Visitant, Soci o Empleat, consultar els productes registrats dins l'aplicació.

Paper dins el treball del usuari: Qualsevol tipus d'usuari poden consultar els diferents productes a partir de tres criteris de cerca: per categoria, per autor i per preu.

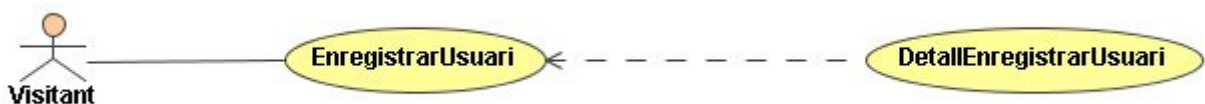
Actors: Visitant, Soci, Empleat.

Precondició: Qualsevol tipus d'usuari.

Postcondició: Es mostra un llistat dels productes cercats a partir dels criteris de cerca.

Descripció: En primer terme, quan es vol cercar informació sobre un producte, es pot cerca per tres criteris diferents: per categoria, per autor o per preu. A continuació es mostrarà el llistat amb els resultats obtinguts.

3.3.4.2 Cas d'Ús EnregistrarUsuari



Funcionalitat: Aquesta opció permet als usuaris de tipus Visitant registrar-se dins l'aplicació. D'aquesta manera els usuaris de tipus Visitant passaran a ser de tipus Soci.

Paper dins el treball del usuari: Qualsevol usuari que consulti la web i no sigui un usuari registrat, podrà registrar-se dins de la mateixa.

Actors: Visitant.

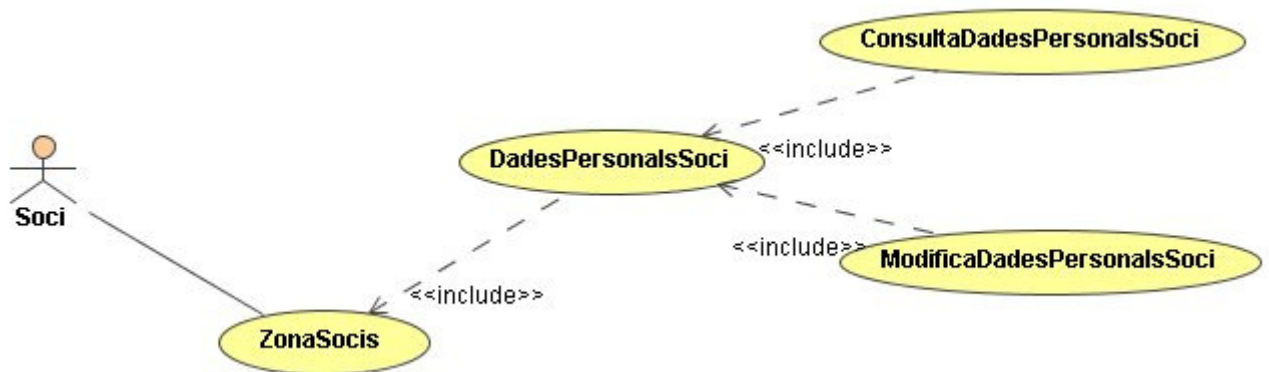
Precondició: Qualsevol usuari que no estigui donat d'alta prèviament a l'aplicació.

Postcondició: Una vegada l'usuari hagi omplert el formulari per el seu enregistrament, es mostrarà un detall de l'enregistrament. Si el procés s'ha efectuat correctament, l'usuari podrà accedir a la *Zona de Socis*; si el procés no s'ha efectuat correctament, l'usuari haurà de repetir el procés corregint els errors mencionat en el detall de l'enregistrament.

Descripció: Qualsevol usuari visitant, podrà registrar-se dins de la tenda virtual a través d'un formulari. En aquest formulari haurà d'omplir informació sobre les seves dades personals. Una ha omplert satisfactòriament el formulari, aquest usuari ja forma part de la llista de Socis.

Observacions: durant el procés d'enregistrament, en el formulari hi haurà camps obligatoris que l'usuari haurà de complimentar; de no esser així no es donarà d'alta en el sistema.

3.3.4.3 Cas d'Ús DadesPersonalsSoci



Funcionalitat: Aquesta opció permet als usuaris de tipus Soci consulta o modificar les seves dades personals com ara, nom, cognoms, domicili, telèfons, email, dades bancàries, etc.

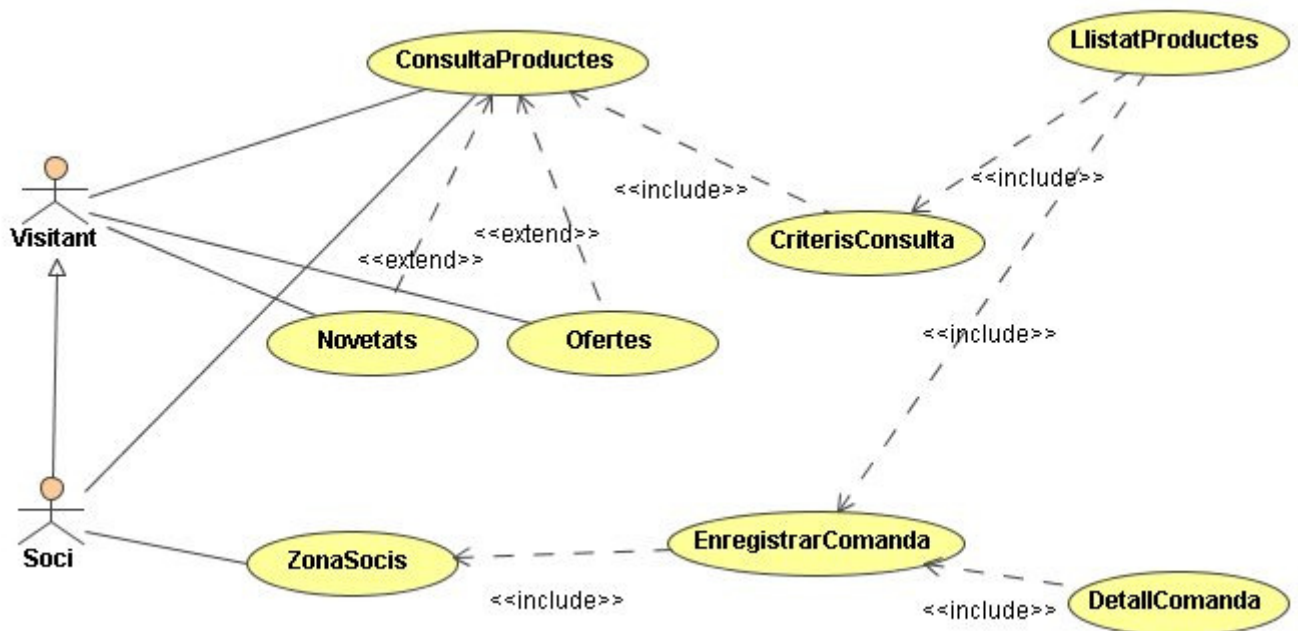
Paper dins el treball del usuari: Permet als usuaris Soci, consultar o actualitzar les seves dades personals.

Actors: Soci.

Precondició: Qualsevol usuari de tipus Soci, que hagi accedit a la *Zona de Socis*, havent-se autenticat prèviament.

Postcondició: Si l'usuari de tipus Soci, ha realitzat modificacions en les seves dades personals, aquestes quedaran actualitzades en la base de dades.

3.3.4.4 Cas d'Ús EnregistrarComanda



Funcionalitat: Aquesta opció permet als usuaris de tipus Soci registrar una nova comanda.

Paper dins el treball del usuari: Qualsevol usuari de tipus Soci i que vulgui realitzar una comanda.

Actors: Soci.

Precondició: Qualsevol usuari de tipus Soci, que hagi accedit a la *Zona de Socis*, havent-se autenticat prèviament.

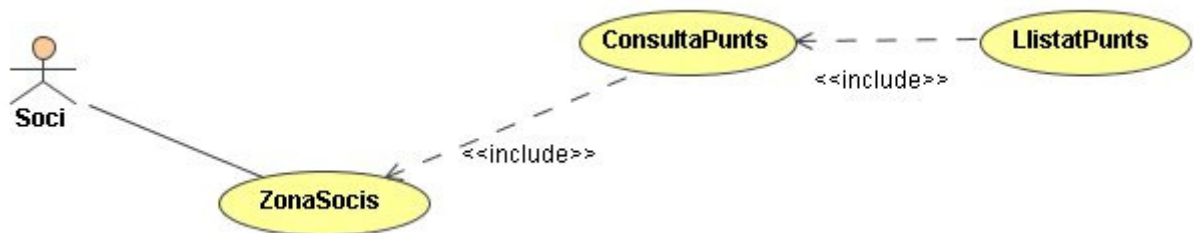
Postcondició: Una vegada l'usuari hagi confirmat la comanda que vol efectuar es mostrarà un detall del total de productes i l'import total.

Descripció: Qualsevol usuari de tipus Soci, que hagi accedit a la *Zona de Socis* i desitgi efectuar alguna comanda, haurà d'afegir els productes al seu *Carrito* particular. D'aquesta manera, quan confirmi la comanda, es comptabilitzarà el total de productes que hi ha al seu carro de la compra, i aquest es tornarà a buidar.

Observacions: alhora d'afegir un producte al carro de la compra, caldrà tenir en compte que hi hagi stock suficient.

En aquest diagrama de casos d'ús, he afegir el de ConsultarProductes, perquè considero que van lligats, ja que pot ser el seu flux lògic. És a dir, un usuari busca un producte concret a través dels criteris de cerca. Una vegada l'ha trobat el selecciona i l'afegeix al seu carro de la compra. Per últim, confirma la comanda.

3.3.4.5 Cas d'Ús ConsultarPunts



Funcionalitat: Aquesta opció permet als usuaris de tipus Soci consultar els punts que tenen acumulats degut a les compres realitzades.

Paper dins el treball del usuari: Es poden consultar els punts disponibles. Amb aquests punts es poden comprar regals del catàleg de punts de la tenda virtual.

Actors: Soci.

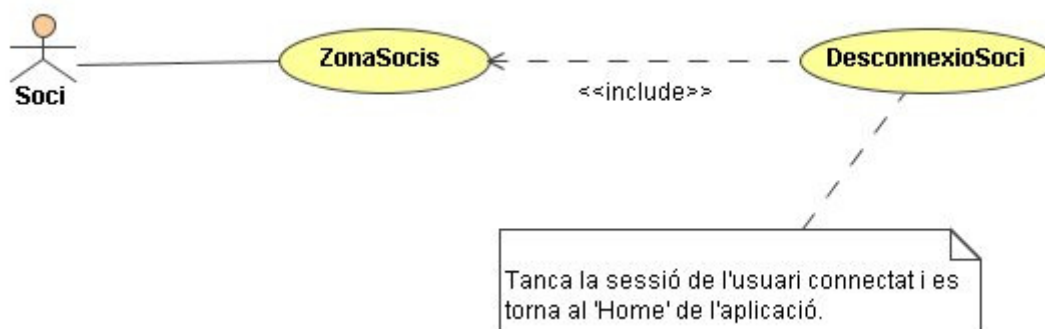
Precondició: Qualsevol usuari de tipus Soci, que hagi accedit a la *Zona de Socis*, havent-se autenticat prèviament.

Postcondició: Es mostrarà el número total de punts acumulats.

Descripció: Qualsevol usuari de tipus Soci, que hagi accedit a la *Zona de Socis* i desitgi consultar els punts acumulats.

Observacions: en l'aplicació actual, els punts sempre són visibles en el marge superior dret de l'aplicació.

3.3.4.6 Cas d'Ús DesconnexioSoci



Funcionalitat: Aquesta opció permet als usuaris de tipus Soci desconnectar-se de l'aplicació.

Paper dins el treball del usuari: A través de l'opció de 'Desconnexió', els usuaris es desconnectaran de l'aplicatiu.

Actors: Soci.

Precondició: Qualsevol usuari de tipus Soci, que hagi accedit a la *Zona de Socis*, havent-se autenticat prèviament.

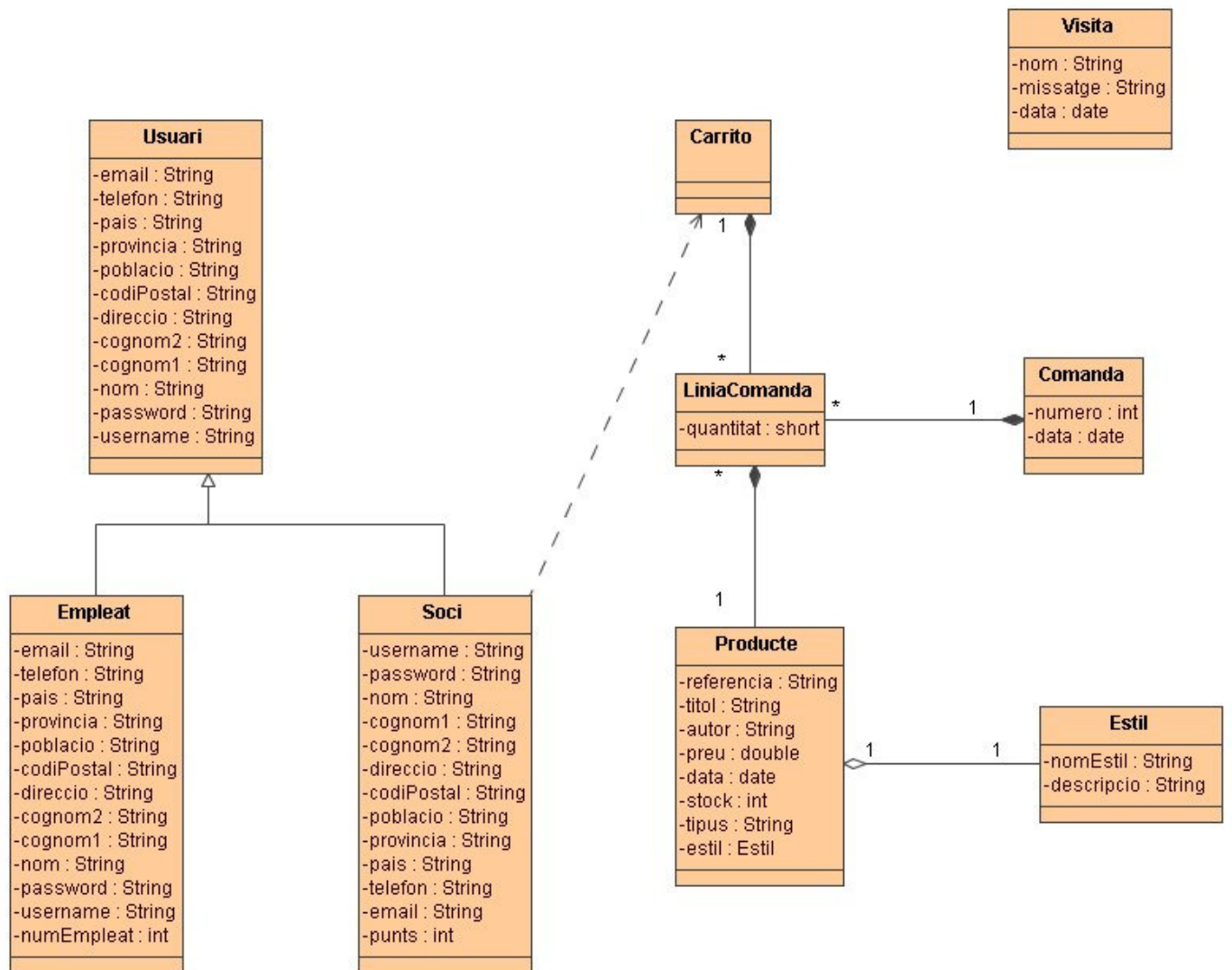
Postcondició: Es retornarà a la pàgina principal de l'aplicació.

Descripció: Qualsevol usuari de tipus Soci, que hagi accedit a la *Zona de Socis* i desitgi desconnectar-se de l'aplicació.

4 Disseny

4.1 Model Conceptual

4.1.1 Diagrama de Classes



Usuari

Aquesta classe defineix el perfil dels usuaris de la nostra aplicació.

A continuació detallo els següents atributs referents a la classe:

username: identificador únic de l'usuari. Pot tenir fins a 10 caràcters alfanumèrics.

password: tindrà una longitud de entre 6 i 10 caràcters.

nom, cognom1, cognom2: cadascun d'aquests atributs podrà tenir un màxim de 25 caràcters.

direcció, població, província: cadascun d'aquests atributs podrà tenir un màxim de 50 caràcters.

país: tindrà una extensió màxima de 25 caràcters.

telèfon: aquest atribut de tipus String emmagatzemarà 9 caràcters numèrics. (També podria contenir caràcters alfanumèrics si és un numero estranger).

email: conté una direcció de correu electrònic. Haurà de contenir el símbol @. Pot tenir una longitud de fins a 50 caràcters.

Soci

Aquesta classe reuneix tota la informació rellevant dels socis "virtuals" de l'aplicació web. Aquests socis queden englobats dins de la classe *Soci*

punts: número enter que indica els punts acumulats per el soci. Aquests punts s'obtenen en cada compra. Quan es crea un Soci per primera vegada, el valor d'aquest atribut és zero. Per a cada 6 € de compra acumularà 1 punt.

Estil

Cada objecte d'aquesta classe determina un tipus d'estil que esta vinculat a cada producte.

nomEstil: correspon al nom de l'estil.

descripció: String que resumeix breument l'estil. Pot tenir fins a un màxim de 50 caràcters.

Producte

Aquesta classe reuneix informació dels productes que es podran vendre dins de l'aplicació virtual.

referència: codi que l'identifica. Haurà de tenir exactament 10 caràcters.

data: l'any d'edició del producte.

stock: número enter que informa sobre la quantitat de producte que queda en el magatzem.

tipus: format de venta del producte. Pot ser CD, MP3 o DVD.

estil: correspon a l'estil del producte.

Carrito

Aquesta classe representa el "Carrito de la compra", és a dir, emmagatzemarà els productes que l'usuari registrat (usuari Soci) vagi seleccionant abans de fer efectiva una compra.

Per tant, podem dir que la classe *Carrito* està composta per un conjunt d'objectes *LiniaComanda*.

afegirProducte: aquest mètode recorre les línies del *Carrito* en busca del producte seleccionat. Si existeix el producte, incrementa la quantitat per aquesta *LiniaComanda* de la comanda. Si no existeix es crearà una nova comanda i s'afegirà una nova *LiniaComanda*.

eliminarProducte: recorre les línies del *Carrito* i eliminarà aquella que conté el producte amb aquella referència.

reassignarQuantitats: aquest mètode rep una llista de quantitats; aquestes quantitats vindran determinades per el número de *LiniesComanda* que tingui el *Carrito* en aquell moment.

getLinies: aquest mètode retornarà les línies comanda emmagatzemades dins la classe *Carrito*.

getPreuTotal: aquest mètode calcularà l'import acumulat en el *Carrito*.

getNumeroLinies: retornarà el número de línies comanda que hi ha al *Carrito*.

buidar: eliminarà tot el contingut del *Carrito* deixant-lo buit.

LiniaComanda

Aquesta classe és la encarregada d'associar productes i quantitats.

create: aquest mètode servirà per crear una nova *LiniaComanda* de un producte amb quantitat 0.

getPreu: retornarà el valor d'aquesta línia de comanda. Aquesta valor es calcularà multiplicant el preu del Producte associat a aquesta línia i a la quantitat.

augmentaQuantitat: aquest mètode incrementarà en una unitat el atribut quantitat.

Comanda

Aquesta classe serà l'encarregada de guardar l'informació sobre les línies de comanda fetes per un soci, el número de comanda i la data de realització.

numero: enter que identifica la comanda.

data: data complerta en la que es va realitzar la comanda.

getLinies: retorna un llista de *LiniesComanda* corresponents a aquesta comanda.

Nota: en el diagrama de classes global, s'ha deixat la classe *Empleat* preparada , per futures ampliacions de funcionalitat.

4.2 Decisions de Disseny

4.2.1 Decisions de l'arquitectura del software

L'arquitectura del software és una descripció dels subsistemes i components d'un sistema software i les relacions entre ells.

Un component és una part encapsulada d'un sistema software; són diferents "peces" que construeixen un sistema software. A nivell de llenguatge de programació es pot presentar com: mòduls, classes, objectes o un conjunt de funcions relacionades.

L'arquitectura és un conjunt de decisions importants sobre:

- L'organització del sistema de software.
- La selecció dels elements estructurals i les interfaces que componen el sistema.
- El comportament dels elements estructurals.
- La composició d'elements en subsistemes més grans.

Els components poden tenir relacions entre ells, que poden ser de dos tipus:

- Estàtiques: tenen que veure amb la col·locació dels components en capes.
Exemple: crides a procediments, accés a variables compartides, etc.
- Dinàmiques: tracten les connexions temporals i les interaccions dinàmiques entre els components.
Exemple: Protocols soci – servidor, protocols d'accés a bases de dades, etc.

Abans de començar el disseny d'un sistema software, tenim que decidir quina serà l'estructura general utilitzada per a obtenir l'arquitectura. Aquesta decisió es basa en les propietats desitjades per el sistema. Les propietats de l'arquitectura, en base als requeriments no funcionals del projecte, en el nostre cas són les següents:

- Canviable:
 - Extensible: noves funcionalitats o millores dels components.
 - Portable: canvis en la plataforma hardware, sistemes operatius, llenguatges, etc.
 - Mantenible: detecció i reparació d'errors.
 - Reestructurable: reorganització de components.
- Interoperable.
- Eficient.
- Fiable.
- Probable.
- Reusable.

Una vegada conegudes les propietats que afecten a les decisions de l'arquitectura, s'esculleixen els patrons arquitectònics que millor s'adaptin a les necessitats del sistema.

Un patró arquitectònic expressa un esquema d'organització estructural fonamental per els sistemes de software. Proporciona un conjunt de subsistemes, especifica les seves responsabilitats e inclou normes i guies per a organitzar relacions entre ells.

Els patrons arquitectònics donen l'esquema general del sistema. Després és necessari adaptar el patró arquitectònic escollit al sistema concret que s'està desenvolupant. Aquesta adaptació es realitza a través de l'utilització de patrons de disseny. El patró arquitectònic que s'utilitza en aquest projecte és:

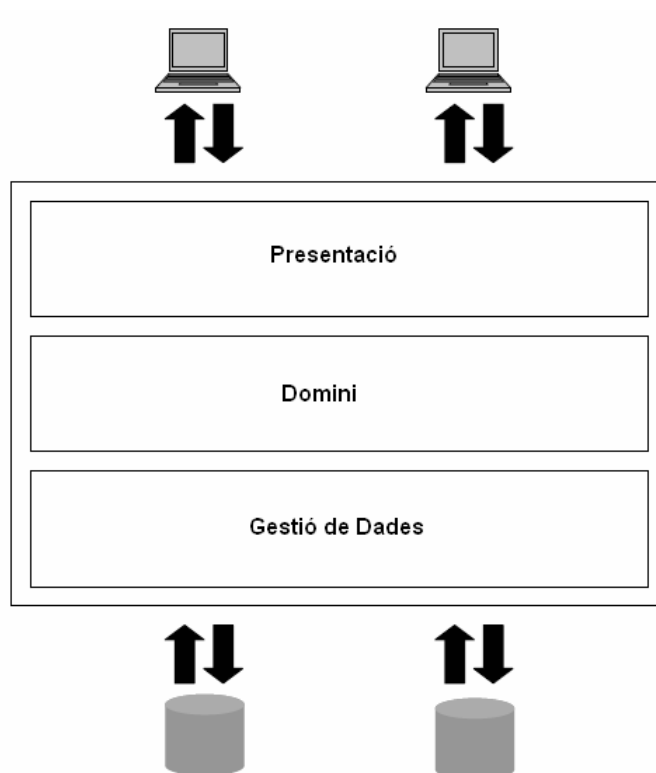
- Arquitectura en tres capes.

Arquitectura de tres capes

S'ha decidit utilitzar un patró arquitectònic en capes, concretament en 3 capes, corresponents a les capes de presentació, domini i gestió de dades.

Les avantatges d'aquesta arquitectura són la reusabilitat (permet reutilitzar capes en diferents contextos), la portabilitat, la canviabilitat (resultat senzill realitzar modificacions gràcies a la estructuració en capes) i que les dependències només siguin locals, és a dir, els canvis en una capa només afectin a aquesta.

A continuació es mostra una captura, de manera esquemàtica, l'arquitectura de tres capes explicat anteriorment:



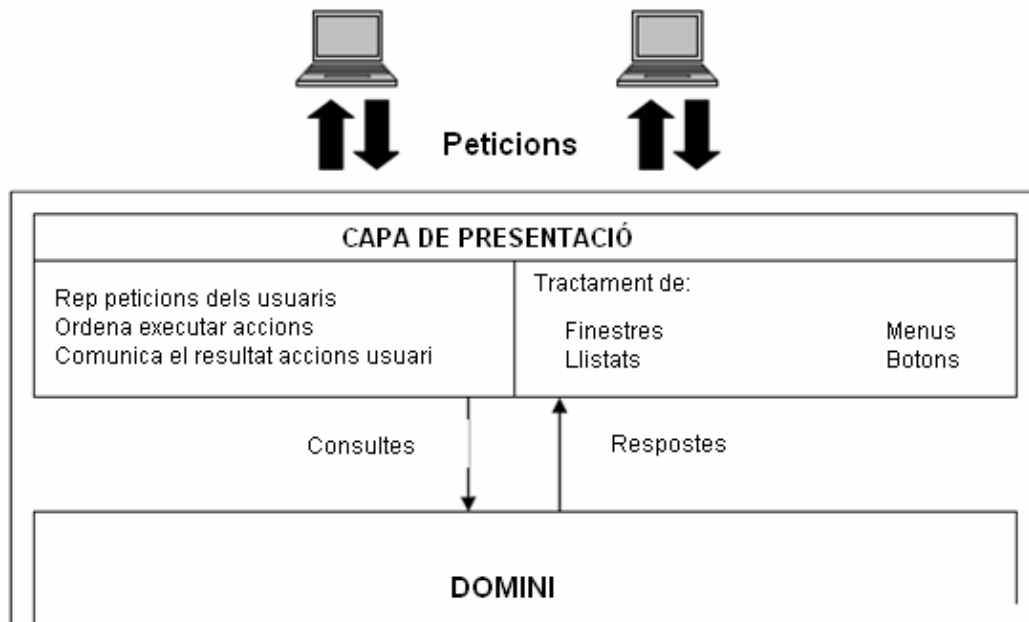
4.3 Capa de Presentació

Aquesta capa s'encarrega del tractament de pàgines, formularis, menús, diàlegs, botons, llistes i altres elements que s'utilitzen per interactuar amb l'usuari. Les funcions principals són:

- Obtenir les peticions dels usuaris.

- Ordenar la execució de les accions.
- Comunicar el resultat de les accions als usuaris.

A continuació es mostra una captura, de manera esquemàtica, la capa de presentació:



El disseny de la capa de presentació és un procés, inclòs en la etapa del disseny de un sistema de software, encarregat de definir:

- Com l'usuari interacciona amb el sistema software (Disseny Extern).
- Com la capa de presentació interacciona amb la capa de domini (Disseny Intern).

Disseny Extern

El disseny extern consisteix en el disseny dels elements que l'usuari veu, escolta i toca al interaccionar amb el sistema.

Per a definir el disseny extern s'ha de tenir en compte els següents aspectes:

- Mecanismes d'interacció: mecanismes per els l'usuari pot fer peticions al sistema.
- Presentació de l'informació: les formes en que es poden mostrar els resultats de les peticions a l'usuari.

Alguns exemples dels components visuals que formen part del disseny extern de l'aplicació són:

- Elements bàsic e imatges.

- Capçalera de l'aplicació.
- Menú.
- Login d'usuari.
- Llistat d'elements.
- Formulari.
- Camps de text.
- Area de text.
- Missatges d'informació.
- Missatges de confirmació.
- Missatges d'error.

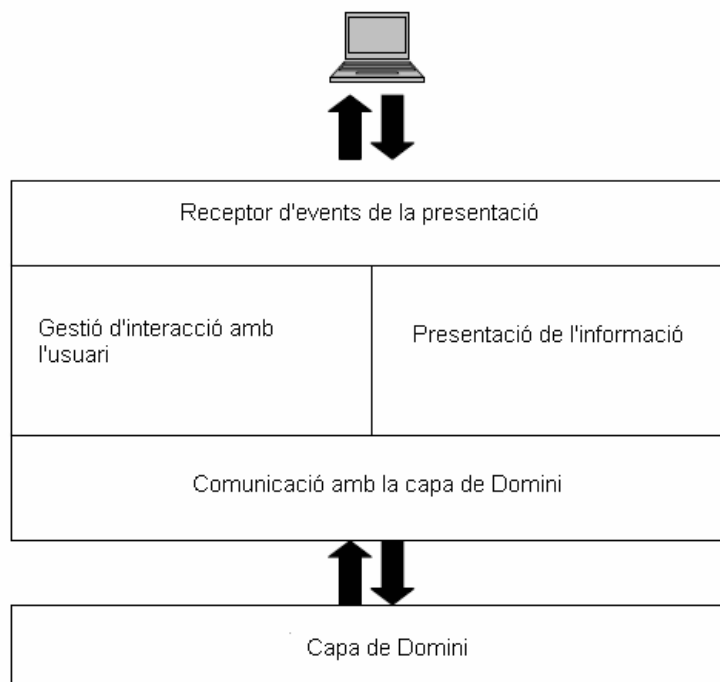
Disseny Intern

El disseny intern de la capa de presentació consisteix en dissenyar els mecanismes que recullen, processen i donen resposta a les peticions de l'usuari.

L'estructura lògica de la capa de presentació es pot dividir en 4 parts:

- El receptor d'events de la presentació.
- La gestió de la interacció amb l'usuari.
- La presentació de l'informació.
- La comunicació amb la capa de domini.

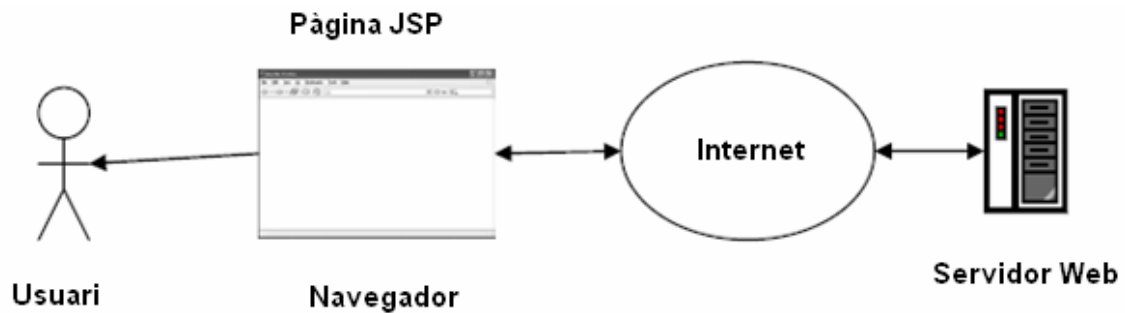
A continuació es mostra una captura, de manera esquemàtica, del disseny intern:



El receptor d'events de la presentació és un conjunt d'interfases gràfiques (GUI) basades en events que permeten la comunicació entre l'usuari i el servidor.

Les interfases gràfiques d'aquest projecte, consisteixen en pàgines JSP que es visualitzen mitjançant un navegador web. És el propi navegador l'encarregat de recollir els events realitzats per l'usuari i comunicar-ho a la capa de presentació.

Exemple:



La gestió de la interacció amb l'usuari s'encarrega de controlar la recepció d'events de presentació del receptor, de processar aquests events i de generar altres events de sistema per els elements que té que processar.

En el cas d'aquest projecte, la gestió d'events la realitzarà el controlador del framework Struts i el servidor d'aplicacions Tomcat.

A l'utilitza una arquitectura en tres capes, és necessari dissenyar la comunicació amb la Capa de Domini que s'encarrega d'enviar events a processar per el sistema i rebre respostes.

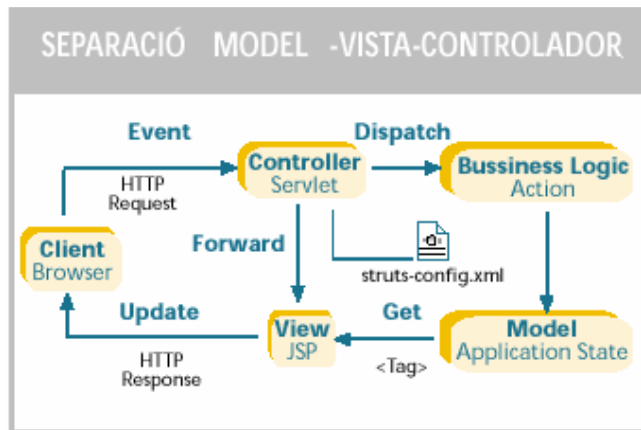
Per el disseny d'aquesta comunicació s'ha decidit utilitzar el patró arquitectònic Model – Vista – Controlador (MVC).

Patró arquitectònic Model – Vista – Controlador

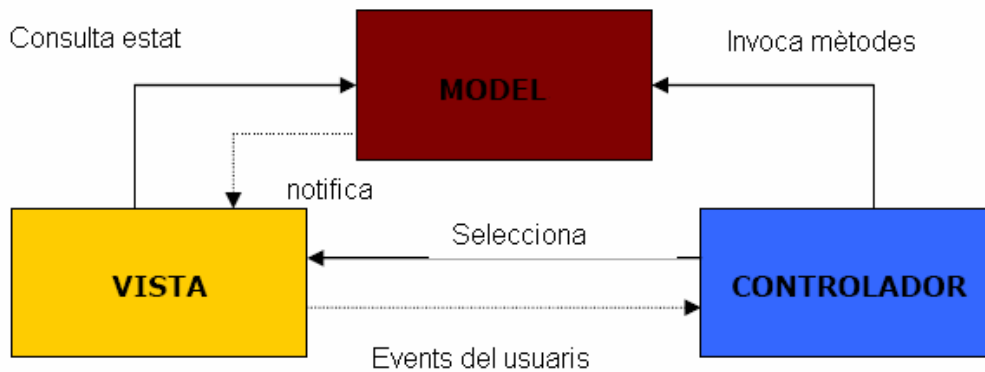
S'ha decidit aplicar el patró arquitectònic Model-Vista-Controlador (MVC) ja que aporta:

- Una baixa dependència entre la capa de presentació i la capa de domini.
- Una alta portabilitat de la capa de presentació.
- Una fàcil reutilització de les vistes i controladors.
- Permet tenir diferents vistes associades al mateix model.

Struts, per naturalesa, implementa el patró Model – Vista – Controlador. A continuació, es mostra un exemple gràfic:



I es desglossa en les tres capes comentades anteriorment tal i com mostra el següent gràfic:



Model:

- Representa les dades de l'aplicació i conté la lògica per accedir i manipular aquestes dades.
- És un conjunt de classes independents de la vista i del controlador.
- Dóna resposta a consultes sobre l'estat.
- El framework Struts no proporciona les classes del model.

Vista:

- Mostra parts del model a l'usuari final.
- Recull peticions dels usuaris i ho envia al controlador.
- És variable en funció del tipus d'usuari (navegador web, dispositiu mòbil, ...).
- En Struts, la vista esta formada per el conjunt de pàgines JSP.

Controlador:

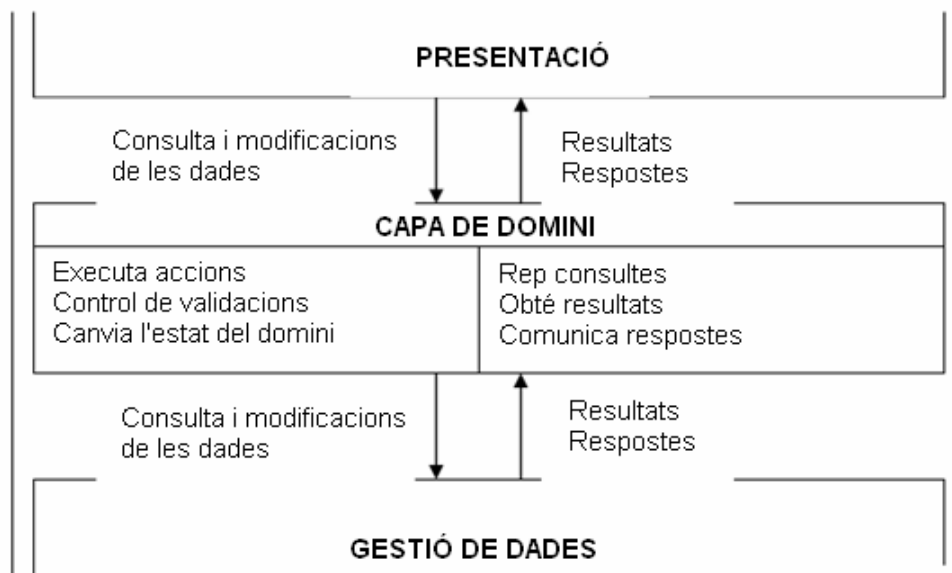
- Interpreta events de l'usuari y actua com a mediador entre el model i la vista.
- Converteix els events de l'usuari en accions que s'executen sobre el model.
- Selecciona les vistes a mostrar.
- En Struts, les Accions (Action) i el fitxer de configuració struts-config.xml realitzen la funció i el treball del Controlador.

4.4 Capa de Domini

L'existència de la capa de presentació permet a la capa de domini ignorar com es presenta l'informació a l'usuari. L'existència de la capa de gestió de dades permet a la capa de domini ignorar on es guarda l'informació. Les funcions d'aquesta capa són:

- Esta al corrent dels events.
- Controla la validesa.
- Canvia l'estat del domini.
- Executa les accions demanades.
- Obté els resultats.
- Comunica les respostes.

A continuació es mostra una captura, de manera esquemàtica, la capa de domini:



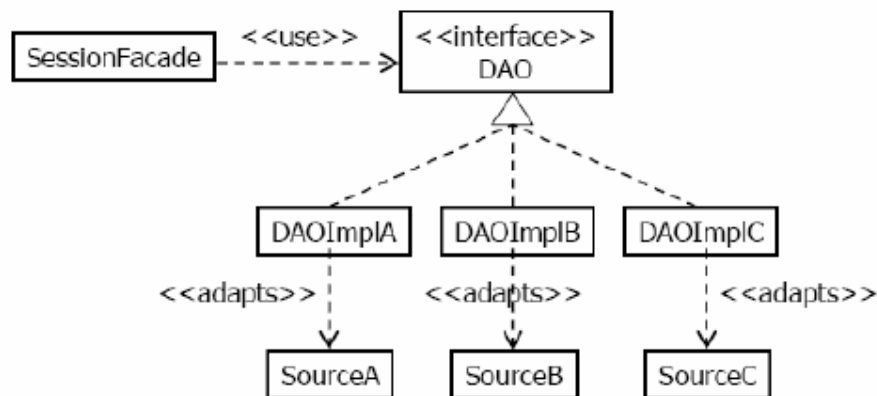
El disseny d'aquesta capa depèn dels patrons de disseny utilitzats. Un patró de disseny ofereix un esquema per definir els subsistemes o components d'un sistema software o les relacions entre elles.

En la nostra aplicació s'ha pensat utilitzar el patró *Data Access Object*.

Patró Data Access Object

Utilitzar un Data Access Object (DAO) per extreure i encapsular tots els accessos a font de dades, aconseguint així desacoblar la lògica de negoci de la lògica d'accés a dades.

A continuació mostro de manera esquemàtica la seva estructura:



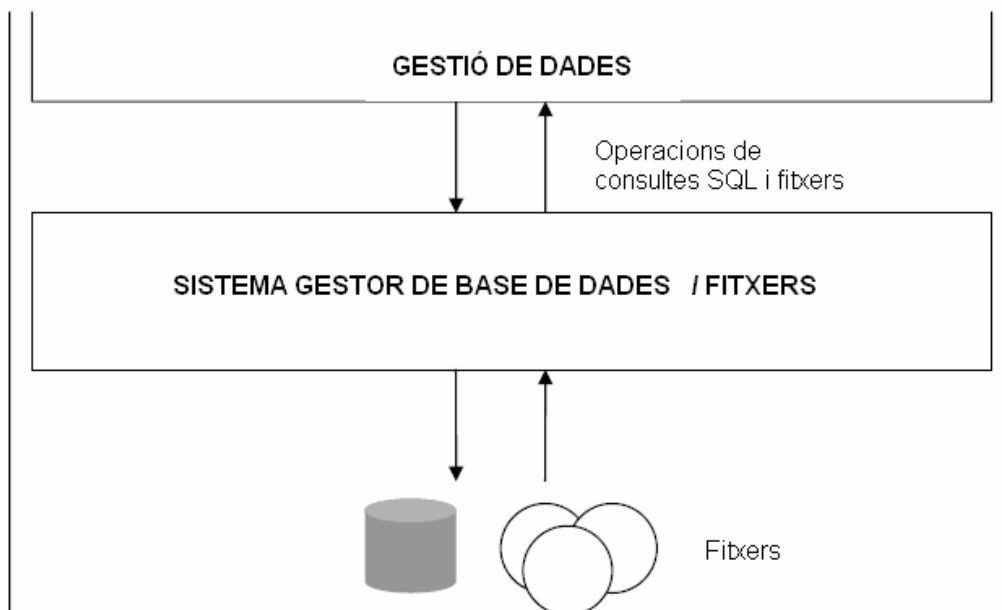
Conseqüències:

- Flexibilitat en l'instal·lació/configuració d'una aplicació.
- Independència de la font de dades.
- Independència del recurs (base de dades relacional, base de dades orientada a objectes, etc.).
- Redueix la complexitat de l'implementació de les funcionalitats (Session Façades).
- Més complexitat en el disseny, degut al conjunt de classes que col·laboren en el DAO i en forma general, cada DAO és utilitzat per modelar una taula del model relacional.

4.5 Capa de Gestió de Dades

Aquesta capa permet que el domini pugui ignorar on està l'informació. Les funcions concretes d'aquesta capa depenen del sistema de gestió de dades que s'utilitzi.

A continuació es mostra una captura, de manera esquemàtica, la capa de gestió de dades:



La persistència és la capacitat que moltes aplicacions requereixen per a emmagatzemar i obtenir dades utilitzant un sistema d'emmagatzament permanent. Aquest és el motiu per el que en la Capa de Gestió de Dades es troben aquells sistemes que necessiten persistència en les seves dades.

4.5.1 Disseny de la Base de Dades

En aquest apartat es mostra el disseny de la base de dades que s'utilitza:

Usuari

Aquesta taula conté informació associada a l'usuari.

Camp	Tipus	PK	Foreign Key
USERNAME	VARCHAR(10)	*	
PASSWORD	VARCHAR(10)		
NOM	VARCHAR(25)		
COGNOM1	VARCHAR(25)		
COGNOM2	VARCHAR(25)		
DIRECCIO	VARCHAR(50)		
CP	CHAR(5)		
POBLACIO	VARCHAR(50)		
PROVINCIA	VARCHAR(50)		
PAIS	VARCHAR(25)		
TELEFON	CHAR(9)		
EMAIL	VARCHAR(50)		

Soci

Aquesta taula conté informació associada a l'usuari Soci.

Camp	Tipus	PK	Foreign Key
USERNAME	VARCHAR(10)	*	
PASSWORD	VARCHAR(10)		
NOM	VARCHAR(25)		
COGNOM1	VARCHAR(25)		
COGNOM2	VARCHAR(25)		
DIRECCIO	VARCHAR(50)		
CP	CHAR(5)		
POBLACIO	VARCHAR(50)		
PROVINCIA	VARCHAR(50)		
PAIS	VARCHAR(25)		
TELEFON	CHAR(9)		
EMAIL	VARCHAR(50)		
PUNTS	INTEGER		

Empleat

Aquesta taula conté informació associada a l'usuari Empleat.

Nota: aquesta taula actualment no s'utilitza, però no s'ha eliminat perquè estigui preparada per a futures ampliacions de funcionalitat.

Camp	Tipus	PK	Foreign Key
USERNAME	VARCHAR(10)	*	
PASSWORD	VARCHAR(10)		
NOM	VARCHAR(25)		
COGNOM1	VARCHAR(25)		
COGNOM2	VARCHAR(25)		
DIRECCIO	VARCHAR(50)		
CP	CHAR(5)		
POBLACIO	VARCHAR(50)		
PROVINCIA	VARCHAR(50)		
PAIS	VARCHAR(25)		
TELEFON	CHAR(9)		
EMAIL	VARCHAR(50)		
NUMEMPLEAT	INTEGER		

Estil

Aquesta taula conté informació relacionada amb els estils de música.

Camp	Tipus	PK	Foreign Key
NOMESTIL	VARCHAR(15)	*	
DESCRIPCIO	VARCHAR(50)		

Producte

Aquesta taula conté informació relacionada amb els productes de la tenda virtual.

Camp	Tipus	PK	Foreign Key
REFERENCIA	CHAR(10)	*	
TITOL	VARCHAR(50)		
AUTOR	VARCHAR(50)		
PREU	DECIMAL (6,2)		
DATA	INTEGER		
STOCK	INTEGER		
TIPUS	CHAR(3)		
NOMESTIL	VARCHAR(15)		ESTIL(NOMESTIL)

Comanda

Aquesta taula conté informació relacionada amb les comandes.

Camp	Tipus	PK	Foreign Key
NUMERO	INTEGER	*	
USERNAME	VARCHAR(10)		SOCI(USERNAME)
DATACOMANDA	DATE		

LiniaComanda

Aquesta taula conté informació relacionada amb les línies de comanda.

Camp	Tipus	PK	Foreign Key
NUMERO	INTEGER	*	COMANDA(NUMERO)
REFERENCIA	CHAR(10)	*	PRODUCTE(REFERENCIA)
QUANTITAT	SMALLINT		

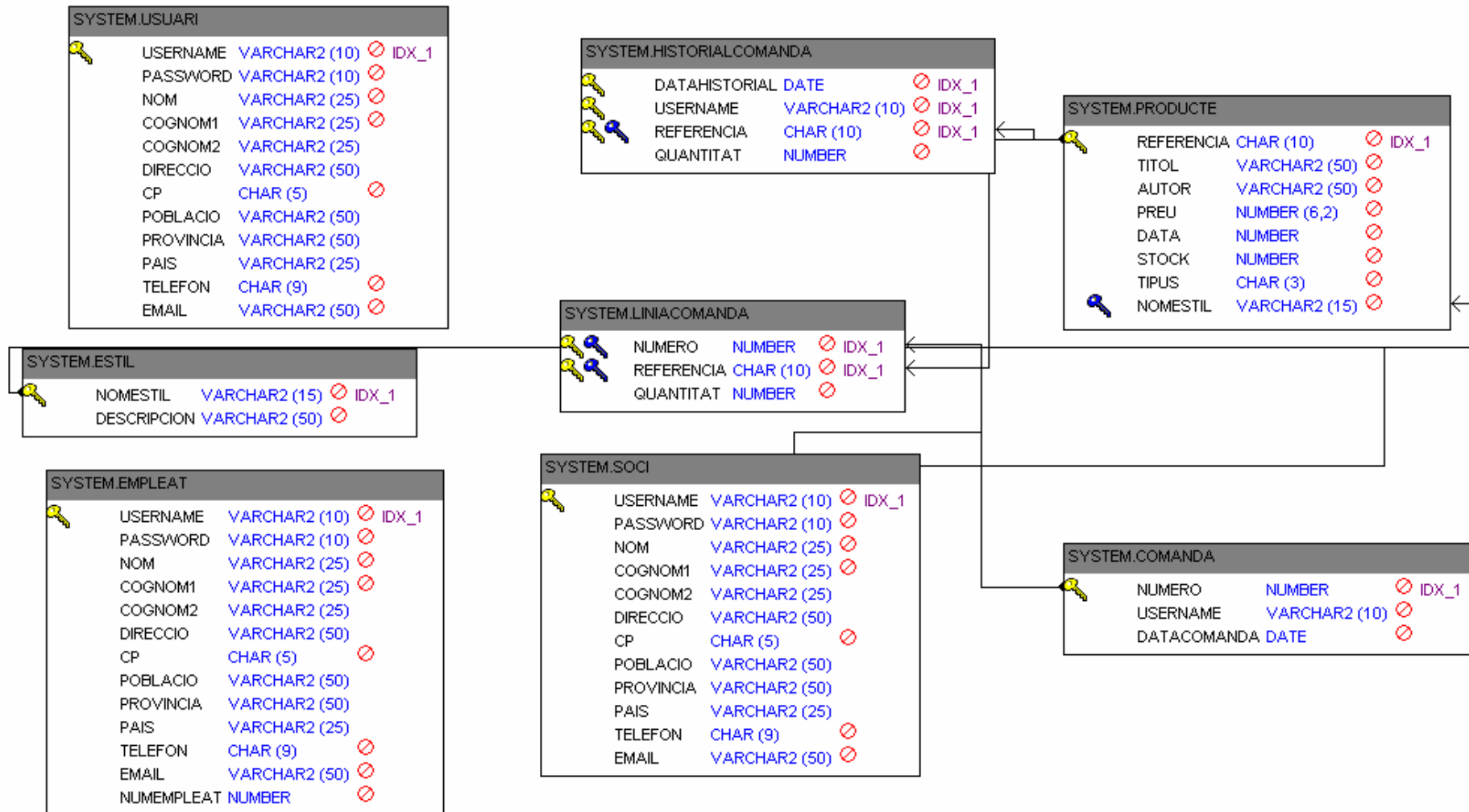
HistorialComanda

Aquesta taula conté informació relacionada amb l'historial de les comandes dels socis.

Nota: aquesta taula actualment no s'utilitza, però no s'ha eliminat perquè estigui preparada per a futures ampliacions de funcionalitat.

Camp	Tipus	PK	Foreign Key
DATAHISTORIAL	DATE	*	
USERNAME	VARCHAR(10)	*	SOCI(USERNAME)
REFERENCIA	CHAR(10)	*	PRODUCTE(REFERENCIA)
QUANTITAT	SMALLINT		

4.5.2 Diagrama ER



4.6 Disseny de les interfícies

L'aplicació s'ha distribuït en 3 parts diferenciades.

La capçalera que correspon a la pàgina Top.html, la part central de la pàgina i el peu de pàgina corresponen a Bottom.html. Aquest cos central, a la vegada, està dividit en 3 parts; el marge esquerre on estan les opcions de menú disponibles, el cos central de l'aplicació es carreguen les pàgines relacionades a les opcions seleccionades i el marge esquerre on es carreguen llistes d'informació.

A continuació mostro captures de pantalla de les pàgines més representatives de l'aplicació fins el moment.

En aquest apartat, es comentarà l'estat actual de l'aplicació, fins a quin punt està implementada.

Dins d'aquest apartat, es pot subdividir en dos apartats, la funcionalitat implementada per els usuaris no registrats, els usuaris visitants, i la funcionalitat implementada per els usuaris registrats, és a dir els usuaris de tipus soci.

4.1.1 Estructura pantalles a través de 'tiles'

En aquest projecte s'utilitza Tiles. Tiles és un component integrat en Struts que implementa el patró Composite View. Permet obtenir una gran potencia i flexibilitat alhora de crear les estructures de les pantalles.

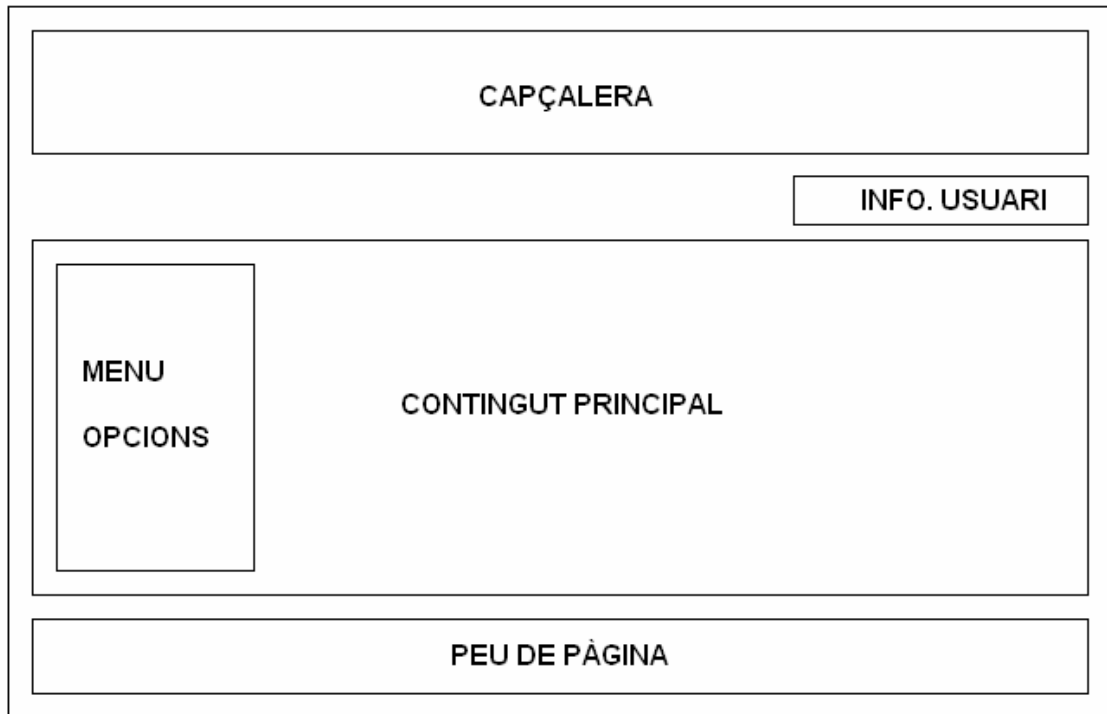
Totes les pantalles segueixen la mateixa estructura:

- **Capçalera:** conté el títol i eslògan de l'aplicació web. També inclou la data del actual.
- **Menú d'opcions:** al marge esquerre de la pantalla, ubicat de forma vertical, hi ha un menú d'opcions que l'usuari pot seleccionar per a realitzar qualsevol acció de les establertes.
- **Contingut principal:** es mostra tot el contingut de la pàgina, ja sigui el contingut per defecte o el resultat d'haver seleccionat alguna opció.
- **Llista productes:** al marge dret de la pantalla, ubicat de forma vertical, hi ha una llista de productes corresponents als productes del mes.
- **Formulari d'autenticació:** ubicat al marge superior dret, formulari que serveix per autenticar l'usuari i accedir a la zona d'accés per a socis.
- **Informació de l'usuari:** ubicat al marge superior dret, es troba informació referent al usuari, com ara el nom i cognoms i el s punts

acumulats. Aquesta informació només està disponible per els usuaris que s'hagin autenticat correctament prèviament.

- **Peu de pàgina:** mostra informació relativa al projecte final de carrera.

La següent figura il·lustra de manera més clara aquesta estructura:



A continuació es mostra de manera esquemàtica les diferents parts de la pantalla. La estructura està dissenyada dins una plantilla a través de tiles.

top

autenticacio

menuOpcions

llista

COS

The screenshot shows the COMUSIC website interface. At the top, there is a blue header with the text "COMUSIC La teva Tenda de Música Online" and a date "20, Maig 2007". To the right of the header is a login form with fields for "Usuari:" and "Contrasenya:", and buttons for "Connectar" and "Corregir". Below the header is a navigation menu on the left with options like "Home", "Cerca per Estil", "Cerca per Artista", "Cerca per Preu", "Fes-te Soci", and "Contacte". The main content area is titled "Novetats Destacades" and features a grid of 12 music album covers with their titles and prices. To the right of the main content is a "TOP 10 Maig" list. At the bottom, there is a footer with the text "TFC J2EE. Comusic. 2ºSemestre 2006-2007, UOC.".

COMUSIC
La teva Tenda de Música
Online

20, Maig 2007

Usuari:
Contrasenya:
Connectar Corregir

Home
Cerca per Estil
Cerca per Artista
Cerca per Preu
Fes-te Soci
Contacte

Novetats Destacades

Its Time
Preu: €15.41

The Annual Spring...
Preu: €23.99

Premonicion
Preu: €8.23

Por la boca vive ...
Preu: €21.85

High School Music...
Preu: €10.99

Greatest Hits
Preu: €9.99

Limon y Sal
Preu: €8.73

The Kind of Magic
Preu: €22.75

Mas Guapa
Preu: €15.59

Pajaros en la cab...
Preu: €6.28

Siempre
Preu: €9.05

Gatos y Palomas
Preu: €8.57

TOP 10 Maig

1. Its Time
2. The Annual Spring 2006
3. Premonicion
4. Por la boca vive el pez
5. High School Musical
6. Greatest Hits
7. Limon y Sal
8. The Kind of Magic
9. Mas Guapa
10. Pajaros en la cabeza

TFC J2EE. Comusic.
2ºSemestre 2006-2007, UOC.

Per a facilitar l'enteniment del mecanisme que utilitza Tiles es mostra un exemple d'una definició utilitzada en el projecte i el procés que s'ha seguit per a arribar a construir-la:

Exemple plantilla per a usuaris visitants:

```
<definition name="templateLayout"
path="/jsp/layouts/TemplatePrincipal.jsp">
  <put name="top" value="/html/Top.html"/>
  <put name="titol" value="- COMUSIC - Principal" direct="true"/>
  <put name="autenticacio" value="/jsp/Autenticacio.jsp"/>
  <put name="menuOpcions" value="/jsp/MenuOpcions.jsp"/>
  <put name="cos" value="/jsp/Novetats.jsp"/>
  <put name="llista" value="/jsp/LlistaTop10.jsp"/>
</definition>
```

Exemple plantilla per a usuaris socis:

```
<definition name="templateLayoutRegistrat"
path="/jsp/layouts/TemplateRegistrat.jsp">
  <put name="top" value="/html/Top.html"/>
  <put name="titol" value="- COMUSIC - Registrat" direct="true"/>
  <put name="autenticacio" value="/jsp/InfoSoci.jsp"/>
  <put name="menuOpcions" value="/jsp/MenuOpcionsSoci.jsp"/>
  <put name="cos" value="/jsp/OfertesSoci.jsp"/>
  <put name="llista" value="/jsp/LlistaTop10Soci.jsp"/>
</definition>
```

El mètode de treball de Tiles consisteix en:

- Classificar les pantalles en grups segons la seva estructura.
- Aïllar les parts comunes en pàgines JSP.
- Definir la pantalla en un fitxer XML.
- Construir una pàgina JSP per cada grup, de manera que defineix l'estructura comuna de totes les seves pantalles.
- Definir cada pantalla en un fitxer XML.

A continuació es mostren algunes captures de pantalla, corresponents a les pàgines més representatives de l'aplicació:

Cerca per Artista:

És un dels exemples de cerca que es pot realitzar. Permet cercar totes els productes a través d'un determinat artista.

Cerca Per Artista

Per cerca un artista, seleccioni la seva lletra inicial en l'abecedari.

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

Amaral
Alejandro Sanz
Andrea Bocelli

Llista productes criteri:

En qualsevol dels tres casos de cerca, es mostrarà una llista amb el resultat del criteri seleccionat. Aquest llista, presenta un sistema de pàgina per facilitar la navegació entre els productes resultants:

Llista Productes Criteri				
Títol	Artista	Data	Suport	Preu
 Zapatillas	Canto del Loco	2006	CD	8.49
 Limon y Sal	Julieta Venegas	2006	CD	8.73
 El tren de los momentos	Alejandro Sanz	2006	CD	10.58
 Gatos y Palomas	Coti	2007	CD	8.57

Anar a la Pàgina: [1](#) [2](#) [3](#)

Al seleccionar qualsevol dels productes mostrats, apareix el detall del mateix:

Detall producte:

Detall Producte



Títol Album:	Zapatillas
Artista:	Canto del Loco
Any Publicació:	2006
Estil:	Pop Rock
Referencia:	d_zapatil
Preu:	8.49 €

Compra:

Format	Preu	Quantitat	
CD	8.49 €	<input type="text" value="1"/>	<input type="button" value="Comprar"/>

En aquesta pantalla es mostra el detall del producte seleccionat. En el cas, que es vulgui compra aquest producte i no estiguem registrats en l'aplicació, o no ens haguem autenticat prèviament ens mostrarà un missatge d'advertència.

Actualització dades personals:

Una vegada l'usuari s'ha registrat i autenticat correctament dins d'aplicació, pot consultar i modificar les seves dades tal i com es mostra:

Dades Personals Soci

Nom*	<input type="text" value="Victor"/>
Primer Cognom*	<input type="text" value="Hill"/>
Segon Cognom	<input type="text" value="Ibar"/>
Direcció	<input type="text" value="Pisuerga 19-21,7ª1º"/>
Codi Postal*	<input type="text" value="08028"/>
Població	<input type="text" value="Barcelona"/>
Província	<input type="text"/>
País	<input type="text" value="Espanya"/>
Telèfon*	<input type="text" value="696125534"/>
E-mail*	<input type="text" value="spyter10@hotmail.com"/>


Informació de la compte

Nom d'usuari de sessió*	<input type="text" value="vhill"/>	El username ha de tenir com a màxim 10 caràcters.
Contrasenya*	<input type="password" value="123456"/>	El password ha de tenir com a mínim sis(6) caràcters de longitud i com a màxim deu (10). Verifiqui que sigui difícil que altres ho adivinin.

Carrito compra:

En el moment que es vulgui comprar un producte, aquest s'afegirà en el carrito de la compra. Dins d'aquest carrito, es podrà triar quatre opcions, recalcular l'import, buidar-lo, seguir comprant o confirmar la comanda.

Carrito Compra



Total Import: 35.59

Títol / Artista	Tipus	Preu	Unitats	Subtotal	Acció
Buleria / David Bisbal	CD	9.00	<input type="text" value="1"/>	9.0	Eliminar
Siempre / Il Divo	CD	9.05	<input type="text" value="2"/>	18.1	Eliminar
Zapatillas / Canto del Loco	CD	8.49	<input type="text" value="1"/>	8.49	Eliminar

Confirmar comanda:

Una vegada es confirma la comanda amb tots els productes seleccionats dins del carrito, es mostra un resum de la comanda feta i el número de punts que s'ha acumulat en la compra.

Comanda Realitzada: N°18



Dades Personals

Nom: Victor
Direcció: Pisuerga 19-21,7°1°

Compres Realitzades

Referencia	Títol	Preu	Tipus
db_buleria	Buleria	9.00	CD
am_pjarocb	Pajaros en la cabeza	6.28	CD
jv_limonys	Limon y Sal	8.73	CD
cl_zapatil	Zapatillas	8.49	CD

Total Comanda: 40.99

En aquesta comanda has acumulat un total de 7 punts.

[Tornar Pàgina Principal](#)

5 Implementació

5.1 Introducció

L'objectiu principal d'aquesta etapa consisteix en implementar el disseny obtingut en la etapa anterior. En la fase de disseny, s'ha construït un sistema però encara no s'havia implementat res. En aquesta etapa, es codifica el sistema amb una estructura concreta per, posteriorment, provar-ho i portar a la pràctica tot el desenvolupament realitzat en les etapes anteriors. En aquest capítol es comenten alguns detalls de d'implementació que són necessaris explicar-los perquè el lector pugui comprendre més concretament el funcionament final de l'aplicació.

Per explicar aquest detalls, aquest document està separat en 4 parts diferenciades.

La primera part, explica les tecnologies utilitzades.

La segona part, tracta de l'estat actual de l'aplicació, fins a quin punt està implementada i quines coses estan pendents.

La tercera part, tracta sobre les decisions de Disseny que s'ha pres.

La quarta i última part d'aquest document, tracta sobre les consideracions que s'ha de tenir en compte per a l'instal·lació de l'aplicació.

5.2 Elecció de tecnologies

En aquest apartat, s'explica quines han sigut les tecnologies que s'han utilitzat per al desenvolupament de l'aplicació.

5.2.1 Sistema Operatiu

El sistema operatiu que s'ha utilitzat tant per el desenvolupament del projecte com per a l'execució de l'aplicació ha sigut **Microsoft Windows XP**.

El motiu principal d'aquesta elecció ha sigut principalment per comoditat, donat que tan el redactat de la memòria com el conjunt d'eines que s'han utilitzat per a l'elaboració del projecte es trobaven disponibles per aquest sistema operatiu.

De totes maneres, tant els llenguatges de programació com el servidor web sobre el qual corre l'aplicació estan disponibles per a múltiples sistemes operatius. Per tant, podem dir que l'aplicació pot córrer de la mateixa manera en un sistema operatiu Windows com en un Linux per el que existeix flexibilitat en quan a la elecció del sistema operatiu.

5.2.2 Servidor Web

Entre els diferents tipus de servidors web que hi ha, s'ha decidit treballar sobre un **servidor web Tomcat**. Concretament sobre la versió Tomcat apache-tomcat-5.5.23.

El fet de treballar amb Tomcat permet que el cost del projecte no augmenti i, al estar recolzat per una gran comunitat d'usuaris, facilita l'obtenció de suport e informació.

5.2.3 Sistema Gestor de Base de Dades

En un principi, es va escollir Oracle 10g, com a sistema gestor de base de dades. Durant l'evolució d'aquesta fase, es va descartar degut a la quantitat de recursos que necessita; sovint es van tenir problemes d'execució degut aquest fet.

Com a possible solució, es va optar per utilitzar **MySQL 5.0**, donat a que necessita molts menys recursos que Oracle i al ésser una eina gratuïta.

5.2.4 Eines de desenvolupament

Com a entorn de programació, s'ha triat **Eclipse**, concretament la versió 3.2. Aquest és un entorn de desenvolupament integrat (IDE), independent de la plataforma, de codi obert, per a crear aplicacions soci de qualsevol mena, la major part del qual ha sigut escrit en Java.

També s'ha utilitzat **Jakarta-Log4j**, projecte de Jakarta Apache, és un sistema de gestió de logs.

5.2.5 Llenguatge en el servidor

Per els llenguatges en la banda servidor, s'ha utilitzat el **framework Struts**, els llenguatges de *script* en el servidor que s'ha utilitzat ha sigut la unió de **servlets** de Java més la tecnologia **JSP** per a la presentació. Concretament, s'ha seguit la versió Servlet 2.0.

Tot el desenvolupament de l'aplicació, s'ha realitzat amb Java i s'ha inclòs la llibreria JSTL que s'ha utilitzat en algunes de les pàgines JSP amb la finalitat de comprendre millor aquestes tecnologies.

5.3 Disseny de pàgines Web

Per el disseny de les pàgines, s'ha intentat separar el codi HTML, de la presentació basat en fulles d'estil. Per això, s'ha utilitzat fulles d'estil CSS que ha permès definir l'estil final dels elements HTML, tals com formularis, títols, paràgrafs, etc. Cal dir que en alguns casos, s'ha afegit codi de presentació dins el codi HTML, per la seva simplicitat.

Cal afegir que en el disseny de les pàgines Web s'han utilitzat patrons de disseny a través de les tiles. A través del fitxer **tiles-defs.xml**, s'han definit tota l'estructura de totes les pantalles web. Aquestes pantalles, es basen en dos patrons o plantilles creades; una per l'aplicació per usuaris visitants i una altre per usuaris registrats.

Aquesta opció, permet estructura molt millor els components visuals de l'aplicació i dona una major flexibilitat alhora de modificacions de l'aplicació.

5.4 Plataforma

Com a plataforma general, s'ha decidit utilitzar la plataforma J2EE, treballant sobre un servidor web Tomcat, implementant l'aplicació amb servlets i utilitzant el framework Struts.

Entre els motius per els que s'han escollit aquests element es troben la gran maduresa que té aquest tipus de tecnologia solucionant molts dels problemes que es pot trobar un dissenyador en el moment de crear una nova aplicació web, així com el fet de ser multiplataforma que permet executar l'aplicació en diferents sistemes operatius.

També, cal afegir que en un principi no es tenia pensat, utilitzar el framework Struts ni **hibernate**, es volia implementar amb servlets i JSP. La recomanació del tutor del projecte, va fer que s'optés per aquesta opció. En un principi una opció molt més costosa, difícil i complexa però per altra molt més gratificant al veure que s'ha après (d'alguna manera) una tecnologia important més.

5.5 Detalls d'implementació

5.5.1 Internacionalització

Una de les característiques de Struts és la capacitat d'internacionalització. Aquesta característica permet a l'aplicació seleccionar l'idioma en el que es mostra l'informació. Així, es mostra una independència entre el codi de les pàgines JSP i l'idioma que es mostra.

El funcionament del mètode d'internacionalització de Struts consisteix en la existència de un conjunt de fitxers de propietats, un per idioma, on apareixen

tots els textos de l'aplicació associats a unes claus comunes en tots els fitxers. És a dir, aquests fitxers són del tipus clau = valor.

En l'aplicació actual, esta preparada per suporta aquesta característica. El fitxer que conté els missatges és el *ApplicationResources.properties*.

5.5.2 Validació de formularis

En diversos punts de l'aplicació, utilitza, sovint, formularis web per a interactuar amb l'usuari. Un dels temes a tenir en compte pel que fa als formularis web és la validació de les dades entrades per l'usuari.

Inicialment, es va pensar a validar els formularis programant les restriccions per a cada formulari. Posteriorment, es va veure que Struts proporcionava un mecanisme de validació de formularis conegut menjo Validator Framework.

Validator Framework

Validator Framework és un mecanisme que facilita la labor de validar formularis amb Struts.

Igual que Struts, Validator Framework suporta internacionalització per a mostrar els missatges d'error a l'usuari pel que, si l'aplicació ja suportava internacionalització, aquesta característica no es perd.

Una dels principals avantatges de la seva utilització és que ja té incorporades moltes regles de validació predefinides per a camps de tipus email, dates, sencers, ... pel que el programador no té perquè implementar-les. A més, el programador pot afegir les seves pròpies regles. A continuació es mostra com s'integra el Validator Framework de Struts en aquest projecte: El primer que cal fer és definir el Validator Framework menjo un plug-in para Struts en el seu fitxer de configuració (struts-config.xml):

```
<plug-in className="org.apache.struts.validator.ValidatorPlugIn">
  <set-property property="pathnames" value="/WEB-INF/validator-
rules.xml,/WEB-INF/validation.xml" />
</plug-in>
```

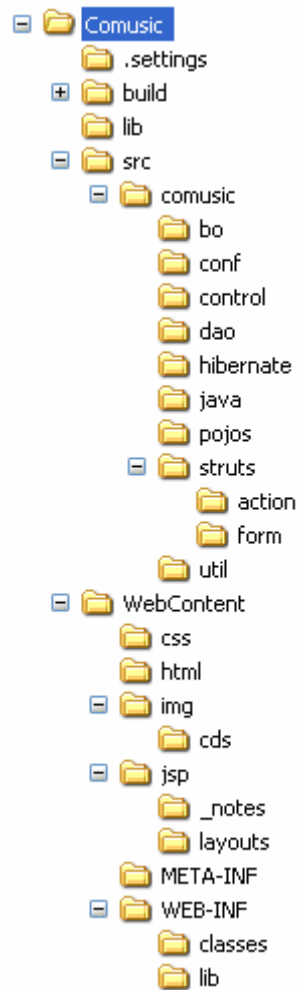
El fitxer validation-rules.xml conté la definició de les restriccions predefinides del validator que seran utilitzades en el fitxer validation.xml. El fitxer validation.xml conté, per a cada formulari que es vol validar, el tipus de restriccions a aplicar en cadascun dels camps.

Perquè els formularis utilitzin aquest tipus de validació, les classes associades als formularis de Struts han d'estendre la classe ValidatorForm que conté el mètode validate():

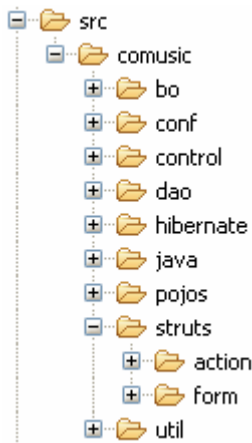
5.5.3 Estructura de directoris

L'estructura de directoris de l'aplicació inclou tant el codi dels servlets que s'executen com el conjunt d'elements que es necessiten per a la correcta visualització (css, jsp, html...).

A continuació es mostra gràficament l'estructura de directoris sencera del projecte *Comusic*:



A continuació es mostra, l'estructura de directoris juntament amb una descripció de la part Model-Controlador:



- **bo:** conté les classes corresponents als *Bussines Objects*.
- **conf:** conté arxius de configuració. En aquesta aplicació conté el fitxer *hibernate.cfg.xml* configurable per a cada connexió de base de dades.
- **control:** conté les classes corresponents al control de l'aplicació. En el nostre cas conté un servlet anomenat *ServletInicialitzacio* que és la primera classe que es crida a l'arrancar l'aplicació; aquest punt es pot veure en el fitxer *web.xml*:

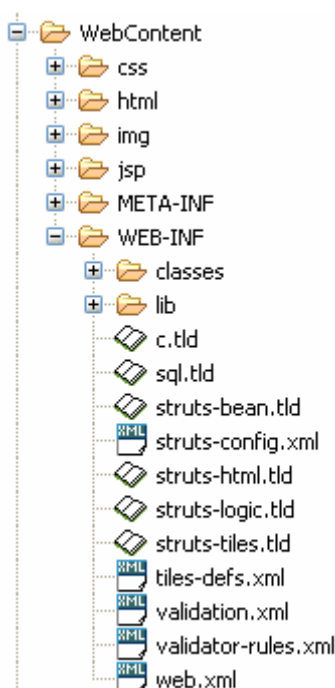
```
<servlet>
  <servlet-name>ServletInicio</servlet-name>
  <servlet-class>comusic.control.ServletInicialitzacio</servlet-
class>
  <load-on-startup>1</load-on-startup>
</servlet>
```

Aquest servlet des de el seu mètode *init()* crida a la classe *FactoriaConfiguracio.java*. Aquesta classe és l'encarregada de configurar correctament tota l'aplicació per el seu correcte funcionament (carregar fitxers, rutes, etc). En el nostre cas, només crida a la ruta d'hibernate.

- **dao:** conté totes les classes que fan accessos a base de dades *Data Access Object* .
- **hibernate:** conté totes les classes que estan mapejades amb la base de dades.
- **java:** conté classes auxiliars que proporcionen funcionalitat a l'aplicació. Són classes de suport, però no interactuen directament amb la base de dades.
- **pojos:** conté les classes corresponents al model, les classes Java corresponents a les taules de la base de dades. Aquest nom correspon a *Plain Old Java Object*.
- **struts:** aquesta carpeta conté, totes les classes corresponents a la lògica de struts. Conté dos subdirectoris:

- **action:** conté totes les classes que hereten de `org.apache.struts.action.Action` o `org.apache.struts.actions.DispatchAction`.
- **form:** conté totes les classes que hereten de `org.apache.struts.validator.ValidatorForm`.
- **util:** conté classes d'utilitat per a diferents classes del projecte com ara constants, conversió de dates, etc.

A continuació es mostra, l'estructura de directoris juntament amb una descripció de la part Vista, la part Web:



5.6 Proves

L'etapa de proves és molt important per a poder obtenir un sistema programari que funcioni correctament. A través de les proves s'ha de poder determinar si la implementació realitzada del sistema satisfà les funcionalitats, les restriccions i els requisits establerts en les etapes d'especificació i anàlisi de requisits. Aquest treball no és senzill ja que cal tenir un bon coneixement dels requisits del sistema desenvolupat.

Les proves no es poden limitar a l'última fase del desenvolupament del sistema una vegada finalitzada la implementació, ja que la detecció d'errors en aquest punt pot perjudicar de certa manera l'èxit del projecte.

Per aquest motiu, s'han anat realitzant proves a mesura que s'implementava, procurant veure que funcionessin tal com s'havia previst. Si el resultat no era

l'esperat, s'ha consultat l'especificació i el disseny del sistema, s'ha modificat la implementació corregint l'error i s'ha tornat a llançar la prova. D'aquesta forma, la tasca de provar el sistema s'ha anat repartint durant l'etapa d'implementació.

Quant als temes tractats en les proves, s'han intentat detectar funcions incorrectes o inexistents, errors en les interfícies, errors en l'accés a les estructures de dades, errors de rendiment i errors en la inicialització i finalització de l'execució.

Cal dir que s'han realitzat diferents tipus de proves. Per una banda el propi desenvolupador de l'aplicació, el qual ha permès trobar i rectificar molts errors. Per una altra banda, han provat l'aplicació altres persones amb diferents nivells de coneixements informàtics, i que a més, no coneixien el desenvolupament de l'aplicació. Aquest fet ha permès trobar errors més evidents i millorar en gran mesura la usabilitat i l'accessibilitat d'un entorn web com és en el nostre cas.

6 Conclusions

Una vegada finalitzat el projecte arriba el moment de treure conclusions. D'una banda, cal comprovar que el sistema desenvolupat compleixi els objectius marcats inicialment.

Per altre banda, arriba el moment de repassar la planificació inicial i comparar el temps estipulat inicialment amb el real. També s'ha realitzat un estudi econòmic del cost del sistema que s'ha desenvolupat.

Abans de donar per finalitzat el projecte, es mostren les línies de futur que poden seguir-se en futurs projectes per a afegir noves funcionalitats al sistema actual. Dins d'aquestes línies de futur, es poden incloure les millores de l'aplicació.

En els següents apartats, es mostra de manera més detallada les conclusions extretes per a cada cas.

6.1 Objectius aconseguits

Després de un bon número d'hores treballant amb el projecte, es pot dir que en línies generals els objectius proposats inicialment s'han aconseguit:

- Subsistema d'usuari visitant: poden consultar la pàgina amb tota la informació disponible. Tenen disponible, tot una sèrie de novetats on poden consultar el detall de cada producte. Poden realitzar cerques sota tres criteris diferents, per estil de música, per artista i per interval de preu. Tenen accés a una llista corresponent als 10 productes del mes. Es poden registrar dins de l'aplicació obtenint així un compte privat.

- Subsistema d'usuari soci: aquest tipus d'usuaris, es poden autenticar i validar dins de l'aplicació a través d'un petit formulari. Una vegada s'han autenticat correctament, accedeixen a un part privada on poden realitzar els tres tipus de cerca comentats anteriorment. Una vegada obtenen el detall d'un producte poden efectuar compres. En tot moment poden consultar l'estat del seu carro de la compra, afegir o eliminar. Finalment poden efectuar la comanda a partir del carro obtingut. També poden actualitzar les seves dades personals.

Cal dir que a banda dels dos sistemes anteriorment descrits, hi havia un tercer, el subsistema d'usuari empleat que ha sigut eliminat de l'especificació per raons d'agenda i temporització.

En un principi s'havia considerat aquesta opció, però s'ha destinat part del temps pensat en implementació en l'aprenentatge i formació de *j2ee*, *struts* i *hibernate*, ja que en un principi es tenia pensat utilitzar *servlets* i *jsps*.

També s'ha destinat una part considerable del temps a la elaboració de la documentació necessària pel desenvolupament. Aquests dos factors, han sigut els motius principals per els quals no s'hagi pogut desenvolupar totes les funcionalitats previstes.

En moltes ocasions, sobretot en l'etapa d'implementació, ha calgut tornar a fer les coses una vegada aquestes ja estaven fetes i funcionaven correctament. El motiu d'aquests canvis és que posteriorment la solució obtinguda a aquests problemes eren millors a les ja existents.

A pesar que l'aplicació no esta del tota acabada i falten alguns punts per polir i millorar, la valoració personal del projecte és força bona.

En quan a l'estat de l'aplicació actualment, considero que esta suficientment encaminada per tal d'acabar-ho de desenvolupar per tal d'obtenir un estat òptim en el qual es pugui posar en producció.

En quan als coneixements, considero personalment molt positivament el grau d'avenç que s'ha tingut en J2EE en general, però concretament en Struts i Hibernate.

6.2 Planificació inicial / real

Una vegada finalitzat el projecte, arriba el moment d'analitzar si el projecte s'ha realitzat segons la planificació inicial.

En aquest apartat es pretén determinar i analitzar les possibles desviacions que s'han produït respecte a la planificació inicial.

El primer que s'ha de fer és obtenir la planificació inicial del projecte i compararla amb les hores reals utilitzades per a desenvolupar-lo. La següent taula mostra una comparativa entre les hores planificades i les hores reals:

Fase	Planificació	Hores Reals
Fase de requeriments	64	72
Descripció funcionalitats	44	52
Especificació del projecte	20	20
Fase d'anàlisi	96	102
Especificació funcionalitats	66	70
Estructura classes i entitats	30	32
Fase disseny	171	180
Diagrames	40	45
Arquitectura	55	60
Interfície gràfica	55	60
Base de dades	21	15
Fase implementació	250	360
Autoformació j2ee	60	80
Codificació aplicació	175	255
Proves	15	25
Fase memòria	47	60
Analitzar canvis funcionals	15	20
Documentació	32	40
TOTAL	628	774

Analitzant els resultats obtinguts es pot observar una desviació de 146 hores pel que fa als establert inicialment que suposa un retard de gairebé un mes i mig.

Analitzant la informació obtinguda en la taula anterior, es pot comprovar que les majors desviacions s'han produït en l'etapa d'implementació i en l'etapa de disseny.

En l'etapa d'especificació el retard prové tant de la definició del model conceptual final com el model de comportament del sistema. En l'etapa d'implementació el retard és fruit del temps utilitzat en l'aprenentatge de les noves tecnologies com de la implementació final de l'aplicació resolent els problemes que anaven sorgint i millorant la implementació ja existent.

6.3 Estudi econòmic

Amb l'objectiu de realitzar un estudi econòmic més detallat sobre el cost total de la realització del projecte s'ha dividit l'estudi en tres parts:

- Cost dels recursos humans.
- Cost dels recursos tecnològics.
- Cost del software utilitzat.

6.3.1 Cost dels recursos humans

Per a calcular el cost dels recursos humans cal conèixer el tipus de perfils que han participat en el projecte, el nombre d'hores reals que ha dedicat cada perfil i el cost per hora de cadascun dels perfils.

Per a la realització d'aquest Projecte de Final de Carrera s'han realitzat treballs propis de dos tipus de perfils:

- Analista: encarregat bàsicament de realitzar l'estudi previ, l'anàlisi de requisits, l'especificació i el disseny.
- Programador: encarregat de realitzar la implementació i les proves.

Una vegada identificats els perfils cal determinar la participació de cadascun en el procés de desenvolupament del sistema. En aquest sentit, la següent taula mostra la distribució dels recursos humans en la diverses etapes del projecte:

Fase	Hores	Analista	Programador
Fase de requeriments	60	60	0
Fase d'anàlisi	102	102	0
Fase de disseny	180	180	0
Fase d'implementació	360	0	360
Fase de memòria	60	30	30

A partir de la distribució en hores dels diferents perfils i del preu per hora d'aquests, es pot calcular el cost dels recursos humans que han participat en el projecte:

Recurs humà	Hores	Cost / Hora	Cost (€)
Analista	372	40	14880
Programador	390	30	11700
TOTAL			26580

6.3.2 Cost dels recursos tecnològics

L'únic recurs tecnològic que s'ha utilitzat durant la realització d'aquest Projecte de Final de Carrera ha estat un PC de treball. Aquest recurs s'ha necessitat durant tot el projecte pel que s'estima que s'ha utilitzat unes 774 hores.

Si considerem que un PC de rènting a la raó de 55 euros al mes, la despesa s'eleva a $4 * 55 \text{ euros} = 220 \text{ euros}$, considerant que el PC s'ha utilitzat durant 4 mesos, corresponents als mesos de Març, Abril, Maig i Juny.

6.3.3 Cost del software utilitzat

Durant la realització del projecte s'ha intentat utilitzar, en la mesura del possible programari lliure per tal de reduir el cost total del projecte.

La major part del programari utilitzat és gratuït. A pesar d'això, en ocasions s'ha utilitzat programari de pagament com en el cas del programari per a escriure la memòria o per al modelatge UML. Per a aquests propòsits s'ha utilitzat el programari Microsoft Office XP valorat en 155 euros i alguna eina de pagament com ara el Magic Draw, encara que tenen períodes d'avaluació gratuïts.

6.4 Línies futures

A partir de l'estat actual de l'aplicació, es pot anar afegint diversa funcionalitat per millorar i oferir més opcions als usuaris. A continuació, proposo alguns punts:

- Un apartat amb els cd's de música més votats per els visitants de la pàgina web.
- La part d'administració dels productes de l'aplicació a través de la web; és a dir, poguer crear nous productes, artistes, etc.
- Una vegada es consulta el detall d'un producte, vincula el producte amb la biografia de l'artista relacionat.
- Permetre que l'usuari pugui consultar el seu historial de comandes, ja sigui per data, nom de producte o import de la comanda.
- Catàleg de regals dels punts.

6.5 Coneixements adquirits

Després de diversos mesos de treball i dedicació a aquest projecte, arriba el moment de recapacitar i intentar treure conclusions sobre tot el que m'ha aportat.

Durant totes les fases del projecte, m'he adonat que t'ensenyen a valorar que és millor pensar abans que actuar sota el primer impuls. He destinat molt temps pensant com seria la millor alternativa, dissenyant l'aplicació amb patrons de disseny i intentat separar al màxim la implementació de les diferents capes. Gràcies a tot això, s'ha obtingut una aplicació flexible, fàcilment modificable, on tots els components estan bé diferenciats i dissenyats.

En quant al factor tecnològic, he de reconèixer que treballar amb Struts no ha estat fàcil, sobretot al principi. En les primeres setmanes, va requerir consultar diversos manuals, exemples i documentació per a entendre totes les seves capacitats. Ara, a dia d'avui, estic orgullós d'haver pres la decisió d'haver treballat amb aquest framework. Treballar amb Struts i Hibernate, m'ha permès comprendre els avantatges i desavantatges que aporta.

En definitiva, estic molt content de tots els coneixements adquirits al llarg de tot el projecte i del resultat final obtingut.

7 Glossari

- Home: opció que permet retorna a la pàgina principal de l'aplicació.
- Cerca per estil: opció que permet a l'usuari cerca els productes a partir d'una llista d'estils. Una vegada s'ha seleccionat l'estil a cercar apareix a través de paginació tots els resultats corresponents.
- Cerca per artista: opció que permet a l'usuari cerca els productes a partir d'una llista d'artistes. En la pàgina apareix un abecedari que es correspon amb la lletra inicial de l'artista. Una vegada s'ha seleccionat l'artista a cercar apareix a través de paginació tots els resultats corresponents.
- Cerca per preu: opció que permet a l'usuari cerca els productes a partir d'un interval de preus. Una vegada s'ha seleccionat l'interval de preus a cercar apareix a través de paginació tots els resultats corresponents.
- Fes-te soci: formulari que permet donar-se d'alta dins l'aplicació com a usuari registrat. Permet obtenir un compte amb informació personal.
- Contacte: opció on es mostra informació relativa a les dades i contacte de l'aplicació per enviar propostes o suggeriments.
- Novetats destacades: correspon als 12 productes amb data de publicació més recent de l'aplicació.
- Llista Top 10: es correspon amb la llista dels 10 productes del mes.
- Dades personals: permet a l'usuari registrat, consultar les seves dades personals i/o modificar-les.
- Carrito compra: opció que permet a l'usuari registrat, veure l'estat de les seves compres.
- Ofertes per a socis: correspon als 12 productes amb el preu més econòmic per als usuaris registrats de l'aplicació.
- Desconnexió: opció que permet tancar la sessió de l'usuari registrat.

8 Bibliografia

Per a la realització d'aquest projecte s'han consultat diverses fonts d'informació que agrupen tant llibres com pàgines web.

En els següents apartats es mostra la bibliografia utilitzada.

8.1 Llibres

- Enginyeria del programari.
Edicions UOC.
- CAVANESS, Chuck. *Jakarta Struts*.
Ed. Anaya Multimedia/ O'Reilly
- Programación Java Server con J2EE Edición 1.3.
Ed. Anaya Multimedia/Wrox.

8.2 Pàgines web

- <http://struts.apache.org>.
- <http://www.hibernate.org>
- <http://www.eclipse.org>
- <http://java.sun.com>
- <http://www.laliluna.de>
- <http://ant.apache.org>
- <http://logging.apache.org/log4j/docs>
- <http://www-306.ibm.com/software/awdtools/developer>
- <http://www.programacion.com>

9 Annex I: Manual d'instal·lació

Un dels objectius al llarg de l'elaboració del projecte, ha estat aconseguir crear una aplicació que es pogués instal·lar fàcilment en un servidor web.

En aquest capítol s'explica quins són els passos a seguir per a poder instal·lar l'aplicació correctament.

9.1 Configuració de l'aplicació

L'aplicació ve comprimida dins d'un fitxer *Comusic.war*. Aquest fitxer conté la mateixa estructura que la que s'ha comentat anteriorment. Conté tot el codi font de l'aplicació.

Alhora d'executar l'aplicació s'haurà de canviar el fitxer de configuració de la base de dades. Aquest fitxer esta ubicat en la carpeta */comusic/conf/hibernate.cfg.xml*. Dins d'aquest fitxer s'haurà de canviar la url corresponent a la ubicació de la base de dades:

```
<property name="hibernate.connection.url">  
jdbc:mysql://localhost:3306/COMUSIC</property>
```

El punt d'entrada a l'aplicació des de qualsevol navegador és:

<http://localhost:8080/Comusic>

9.2 Configuració de la Base de Dades

Juntament amb l'aplicació es lliure un altre fitxer anomenat *BD_Comusic.rar*.

Aquest fitxer comprimit , conté un script per la creació de la base de dades, fitxer anomenat *script_COMUSIC_MySql.sql*. També conté una carpeta COMUSIC on dins hi ha codi SQL corresponen a les insercions de les taules *soci*, *estil* i *producte*.