

PFC PLANA WEB D'AJUDA A LA DECISIÓ EN LA COMPRA DEL SUPERMERCAT

Índex

1 Introducció.....	5
1.1 Pròleg.....	5
1.2 Motivació.....	6
1.3 Definició del problema.....	7
1.3.1 Article web relacionat (any 2008).....	7
1.4 Objectiu del projecte.....	9
1.5 Estructura de la memòria.....	10
2 Anàlisis de requisits.....	11
2.1 Requeriments funcionals.....	11
2.2 Requeriments no funcionals.....	12
2.2.1 Requisits d'aspecte (look&feel).....	12
2.2.2 Requisits d'usabilitat	12
2.2.3 Requisits de rendiment.....	13
2.2.4 Requisits de seguretat	13
2.2.5 Requisits legals.....	14
2.2.6 Mantenibilitat.....	14
2.2.7 Restriccions de disseny.....	15
2.2.8 Requisits hardware.....	15
2.2.9 Requisits software.....	15
3 Especificació.....	16
3.1 Diagrama dels casos d'ús.....	18
3.2 Registrar-se al sistema.....	20
3.3 Alta d'usuari privat.....	21
3.4 Alta llista de la compra.....	22
3.5 Eliminar llista de la compra.....	23
3.6 Modificar nom de la llista de la compra.....	24
3.7 Afegir producte a la llista de la compra.....	25
3.8 Eliminar producte de la llista de la compra.....	26
3.9 Predeterminar llista de la compra.....	27

3.10 Carregar llista de la compra al carretó.....	28
3.11 Llistat dels productes del sistema.....	29
3.12 Afegir producte al carretó.....	30
3.13 Eliminar producte del carretó.....	31
3.14 Buidar carretó.....	32
3.15 Finalitzar comanda.....	33
3.16 Llistar històric de llistes impreses.....	34
3.17 Comparativa de les mitjanes de preus dels supermercats.....	35
3.18 Comparativa de les mitjanes de preus per secció.....	36
3.19 Comparativa de la trajectòria econòmica de l'usuari.....	37
3.20 Cerca de productes	38
3.21 Imprimir llista.....	39
3.22 Guardar en PDF una llista.....	40
3.23 Accés (“login”) al sistema.....	41
3.24 Sortida (“logout”) del sistema.....	42
3.25 Entrada (“login”) al panell de control del sistema.....	43
3.26 Sortida (“logout”) del panell de control del sistema.....	44
3.27 Afegir nova secció al sistema.....	45
3.28 Eliminar secció del sistema.....	46
3.29 Modificar el nom d'una secció del sistema.....	47
3.30 Modificar usuari registrat.....	48
3.31 Eliminar usuari registrat.....	49
3.32 Estadístiques del sistema.....	50
3.33 Alta d'un nou producte al sistema.....	51
3.34 Baixa d'un producte del sistema.....	52
3.35 Modificació d'un producte del sistema.....	53
3.36 Llistat dels productes.....	54
4 Avaluació tecnològica.....	55
4.1 Arquitectura client/servidor.....	55
4.2 HTTP.....	58

4.3 Llenguatges de programació.....	60
4.3.1 HTML.....	60
4.3.2 PHP.....	62
4.3.3 Javascript.....	65
4.3.4 AJAX.....	67
4.3.5 XAJAX.....	69
4.3.6 CSS.....	71
4.3.7 XML.....	72
4.3.8 SQL.....	75
4.4 Servidor web.....	77
4.4.1 Apache.....	78
4.5 Sistema Gestor de Base de Dades.....	79
4.5.1 MySQL.....	79
5 Disseny.....	83
5.1 Arquitectura del sistema.....	83
5.2 Disseny de l'aplicació.....	84
5.2.1 Patró DataMapper.....	84
5.2.2 Patró Adapter.....	86
5.3 Diagrama conceptual.....	88
5.4 Disseny de la web.....	89
6 Implementació.....	91
7 Planificació i costos.....	100
7.1 Diagrama de Gantt.....	100
7.2 Cost del projecte.....	102
8 Conclusions i possibles extensions futures.....	105
9 Bibliografia.....	108
10 Annex.....	110
10.1 Glossari.....	110
10.2 Manual d'usuari.....	114

1 Introducció

1.1 Pròleg

A dia d'avui, 22 de març de 2009, i des de farà aproximadament 1 any, tot el món es troba immers en una gran crisi econòmica, es moment de plantar-li cara i d'aprofitar els recursos que tenim al nostre abast per a sortint-se'n quant abans millor d'ella.

Des del meu projecte, el que puc aportar a aquesta situació actual, és una aplicació web amb el propòsit d'estalviar temps i diners a tota persona interessada gràcies a Quamarket, una proposta de codi obert, on l'objectiu és obtenir la llista de la compra setmanal però amb la diferència que es cercarà dins d' una gran base de dades els productes més econòmics d'entre tots els presents.

Més concretament, es realitzarà una plana web dinàmica, aprofitant les tecnologies actuals com ara la web 2.0, AJAX i Javascript entre d'altres per a assolir l'objectiu.

Òbviament per lo citat, aquest projecte serà realitzat en la seva totalitat per software lliure, aprofitant, si escau , llibreries i aplicacions de codi obert ja realitzades per altres usuaris.

1.2 Motivació

El perquè he escollit aquest projecte, es podria dir en poques paraules: soc un apassionat de tot el que funciona via Internet. Des del principi que va aparèixer aquesta tecnologia que no hi ha hagut dia que no l'hagi utilitzada. Es impressionant poder comunicar-nos d'una punta del món a l'altre en dècimes de segon. Ja des de petit que em passava el dia fent planes web's (personals) en comptes d'enganxar-me a videojocs per exemple. Des de que vaig cursar la assignatura DSBW (Disseny de Sistemes Basats en la Web) i PXC (Projecte de Xarxes de Computadors) aquí la FIB, m'ha interessat encara més aquest tema. Cal dir que professionalment encara no m'hi he dedicat però no ho descarto en un futur proper.

Espero que aquest projecte em serveixi per assolir els coneixements conceptuals adquirits a la FIB aplicats al món web, i donar el pas cap al món laboral.

1.3 Definició del problema

Les possibilitats de les persones en quant a realitzar compres via Internet avui en dia comença a ésser notable. No obstant, si s'inicia la compra en un supermercat, es ben segur que hi trobarem productes més cars que en altres establiments i viceversa. Cal dir també, que avui en dia la gent no disposa de temps en excés com per anar de supermercat en supermercat mirant on pretén trobar l'article al millor preu, i també dir que la gent es reticent a comprar altres marques dels seus productes habituals.

Si es seguís rigorosament la compra de l'article més barat en diferents súpers, a final d'any l'estalvi seria notable. Segons un estudi que cito a continuació, l'estalvi global podria arribar a ser força considerable.

1.3.1 Article web relacionat (any 2008)

L'Organització de Consumidors i Usuaris (OCU) ha realitzat, un any més, l'enquesta de preus de supermercats. Durant dues setmanes s'han visitat 801 establiments de 53 ciutats i d'altres per Internet. S'han recollit prop de 80.000 preus de 126 productes diferents en establiments de diferents tipus: supermercats, hipermercats i botigues de descompte. Aquesta gran enquesta permet fer valoracions significatives sobre les polítiques de preus dels diferents establiments comercials.

En el seu estudi, la OCU analitza els preus dels supermercats mitjançant l'elaboració de dues "Cistelles" diferents. La *Cistella Tipus*, formada a la vegada per la *Cistella de Frescos* i la *Cistella d'Envasats*, està composta pels productes de consum més habituals que compra una família mitjana espanyola. Conté 146 productes concrets (dividits en 18 seccions), tots ells de marques capdavanteres en

el seu sector, ponderats d'acord al pes de cada sector en l'enquesta de pressupostos familiars elaborada per la INE (Institut Nacional d'Estadística). I la *Cistella Econòmica*, la qual està dirigida a aquells consumidors, cada vegada més nombrosos, que opten per comprar els productes més barats, amb independència de la seva marca. Aquesta cistella està integrada per 80 productes definits pel seu contingut, format i volum, (per exemple un litre de llet sencera en envàs de TeTra Brik) però no per la seva marca. Pertanyen a tres grups diferents: aliments envasats, begudes i articles de drogueria i higiene.

Com a resultats s'obtenen unes diferències considerables de preu depenent de l'establiment on es compri. Aquestes notables diferències de preus entre establiments permeten segons la OCU, realitzar importants estalvis pel consumidor en funció de l'establiment o cadena que triï per a realitzar la compra. Per a la *Cistella Tipus*, l'estalvi mig entre el més barat i el més car dels establiments és de 617 euros a l'any (menor que en el 2007 que va ser de 673 euros). A Madrid és possible estalviar fins a 1.565 euros, el màxim estalvi possible a nivell nacional, un 37% sobre l'import total de la compra. En moltes capitals de província és possible estalviar més de 1.000 euros a l'any: Bilbao, Granada, Murcia, Palma de Mallorca i Oviedo. Per contra, l'estalvi menor possible es troba a Cuenca i Puertollano, amb un estalvi entorn els 110-120 € anuals.

Per a la *Cistella Econòmica* l'estalvi entre l'establiment més car i el més barat és encara major: El màxim possible seria de 2.296 euros a l'any. Aquestes grans diferències permetrien que amb el que gasta un consumidor durant un any en l'establiment més car, es pogués comprar durant tot un any i fins al pont de la Constitució de l'any següent, en l'establiment més barat.

Font:

<http://www.ocu.org/ahorro-e-inversion/es-posible-ahorrar-mas-de-1-500-euros-anuales-en-la-cesta-de-la-compra-s406294.htm>

1.4 Objectiu del projecte

Quamarket neix amb l'objectiu d'alleugerir el problema que s'acaba de presentar, cercar, d'entre els productes introduïts al sistema, els habituals de cada usuari al millor preu i imprimir la llista de la compra. La plana serà totalment dinàmica i es construirà en la seva totalitat mitjançant software lliure.

Es realitzarà un projecte complet, és a dir, passant per totes les etapes: Anàlisi de requisits, especificació, disseny, implementació, testing i creació del entregable.

També és un objectiu de projecte, aprofundir en les diferents tecnologies lliures actuals de que disposem i el de realitzar un projecte complet d'inici a fi, posant-li finalment un preu al producte resultat.

1.5 Estructura de la memòria

La memòria s'ha organitzat de la següent manera:

En el primer capítol es farà una petita introducció, la qual contindrà un pròleg, una motivació de perquè he escollit aquest projecte i no un altre, una posada en situació de que pot aportar aquest projecte i els objectius que he marcat.

En el segon capítol es farà el anàlisis de requisits del projecte, concretament la presa de requeriments funcionals i no funcionals.

En el tercer capítol es detallaran els diferents casos d'ús del sistema.

En el quart capítol es farà una avaluació tecnològica explicant tota la tecnologia utilitzada.

Un cinquè capítol on es detallarà la part de disseny, quins patrons s'utilitzaran i perquè.

Un sisè capítol on es comentarà breument la implementació de les diferents parts principals del sistema.

En el setè capítol es realitzarà un petit estudi de planificació i costos reals que implicaria el projecte si sortís al mercat.

En el vuitè capítol es descriuen les conclusions extretes del projecte i les possibles futures ampliacions a partir d'aquest.

Finalment, també adjuntaré la bibliografia que he precisat i un annex amb un glossari i un manual d'instal·lació de l'aplicació.

2 Anàlisi de requisits

Aquest apartat pertany a l'enginyeria del software, una branca fonamental en la informàtica que pretén construir software de qualitat mitjançant metodologies i tècniques ja provades. En concret, durant l'etapa d'anàlisi de requisits es defineix que es el què haurà de fer el sistema (requisits funcionals). Es una part molt important a tenir en compte ja que normalment, el client no sap exactament el que vol i som nosaltres qui ho hem d'anar descobrint en aquesta etapa (si no volem després tenir que repetir feina). La feina d'aquest apartat es doncs, completar possibles requisits incomplets, ambigus o contradictoris.

Es començarà descrivint els requeriments funcionals, aquells independents de la tecnologia a usar i es seguirà pels no funcionals (els que ja depenen de la tecnologia).

2.1 Requeriments funcionals

Els requeriments funcionals que oferirà el sistema seran:

- Registre d'usuaris
- Registre d'usuaris privats
- Gestió de les llistes de la compra
- Altes i baixes de productes dins del carretó
- Llistat dels productes del sistema per seccions
- Reiniciat de la llista de la compra
- Finalitzar una comanda
- Historial de comandes realitzades
- Visualitzar comparatives estadístiques entre supermercats
- Cerca de productes

- Gestió de la llista de la compra
- Accés al panell de control del sistema
- Gestió de les categories del supermercat (seccions)
- Gestió dels productes del sistema
- Gestió dels socis del panell de control
- Gestió dels usuaris registrats
- Estadístiques del sistema (en el panell de control)

2.2 Requeriments no funcionals

A continuació s'explicaran els requisits no funcionals, o dit d'altra manera, aquells que s'imposen alhora de realitzar un projecte. Els requeriments no funcionals els podríem dividir en:

2.2.1 Requisits d'aspecte (look&feel)

El sistema ha de ser fàcil d'utilitzar, intuïtiu per a qualsevol persona tingui l'edat que tingui i independentment de la seva perícia informàtica.

2.2.2 Requisits d'usabilitat

Facilitat d'ús

Es considera objectiu primordial el fer de la navegació pel sistema una tasca fàcil, precisa i intuïtiva. La navegació per la web serà molt dinàmica i fluida, es per tant, que l'eficiència serà també un aspecte clau.

Llegibilitat

La informació mostrada en el portal web haurà de ser molt llegible, es a dir, intentar evitar llargs paràgrafs i densitat d'informació que pugui crear sensació de cansament a l'usuari.

Compatibilitat

Al tractar-se d'un sistema web, la compatibilitat haurà de ser total respecte als navegadors de major pes social, com Mozilla Firefox, Internet Explorer, Opera, Safari o Google Chrome, aconseguit així un bon *Cross Browsing*. Per tal de complir aquesta precondició es programarà tot el sistema seguint els estàndards i pautes de la W3C (World Wide Web Consortium).

2.2.3 Requisits de rendiment

Temps de resposta per transacció

El temps estimat de resposta per transacció ha de ser estrictament entre 1 i 2 segons. En cas de superar aquest interval, s'afegiran barres de progrés per informar a l'usuari.

Accessos a la base de dades

El portal web serà capaç de gestionar un total de 200 consultes per segon a la base de dades del sistema.

2.2.4 Requisits de seguretat

El sistema es prepararà per a evitar que s'insereixi codi maliciós (p.ex: *sql injection*). Tanmateix, qualsevol petició que elimini informació de la base de dades se li notificarà a l'usuari amb una finestra de confirmació.

Al moment de donar-se d'alta un usuari, aquest haurà d'introduir una contrasenya. Si aquesta no arriba a un mínim de seguretat establerta no deixarà continuar el registre. Concretament, les contrasenyes hauran de:

- Tenir una longitud mínima de 6 caràcters
- Assolir el 30% en el càlcul del nivell de seguretat. Per arribar a aquests mínims s'hauran de barrejar lletres i nombres així com majúscules i/o símbols no alfanumèrics.

A més a més, a l'accedir al panell de control del sistema s'haurà d'inserir unes lletres/nombres que es veuran en una imatge (també anomenat sistema de *captcha*) per tal d'evitar que possibles *bots* de la xarxa intentin abordar el sistema. Tot i així, si es fracassa tres cops l'ingrés (*login*), es bloquejarà la IP de l'usuari durant 15 minuts.

2.2.5 Requisits legals

L'aplicació desenvolupada serà de codi obert, és a dir, qualsevol persona podrà obtenir, modificar i redistribuir el codi font del programa, amb el fi d'adaptar-lo a les serves necessitat, i millorar-lo.

2.2.6 Mantenibilitat

Codi estructurat i comentat

El codi de l'aplicació web haurà d'estar degudament estructurat i comentat, amb la finalitat de facilitar la llegibilitat i la seva interpretació, de cara a futures modificacions i/o ampliacions.

Documentació relativa al disseny i implementació de l'aplicació

Es presentarà un document que contindrà l'especificació software del disseny i implementació del sistema, a més de les justificacions de les diferents decisions preses.

Documentació relativa a les diferents bases de dades

S'adjuntarà un document que inclourà l'especificació i el diagrama de classes de les diferents bases de dades usades, a més de documentació relativa al *SGBD* utilitzat.

2.2.7 Restriccions de disseny

Requisits de la plataforma

La web serà accessible des de qualsevol ordinador mitjançant una xarxa convencional i el seu navegador preferit independentment del sistema operatiu. Tot i no programar res en Flash, l'equip de l'usuari haurà de tenir instal·lat Adobe Flash Player per tal de poder utilitzar el 100% de les funcionalitats de la web, ja que en la realització de l'aplicació web, s'utilitza un generador de gràfics que conté una pel·lícula Flash (*open XML/SWF Charts*).

Restriccions del llenguatge

El llenguatge utilitzat serà exclusivament PHP + AJAX + Javascript + CSS per la part de la web i SQL per al servidor de la base de dades.

2.2.8 Requisits hardware

Al no precisar potència gràfica ni funcionalitats avançades, el sistema pot instal·lar-se a qualsevol ordinador superior a un pentium II amb connexió a Internet si el que es pretén es oferir el servei a més persones.

2.2.9 Requisits software

Els ordinadors dels usuaris hauran de disposar d'un navegador web, sigui el que sigui ja que com s'ha comentat abans un requisit es que serà compatible amb la totalitat dels navegadors estàndards actuals.

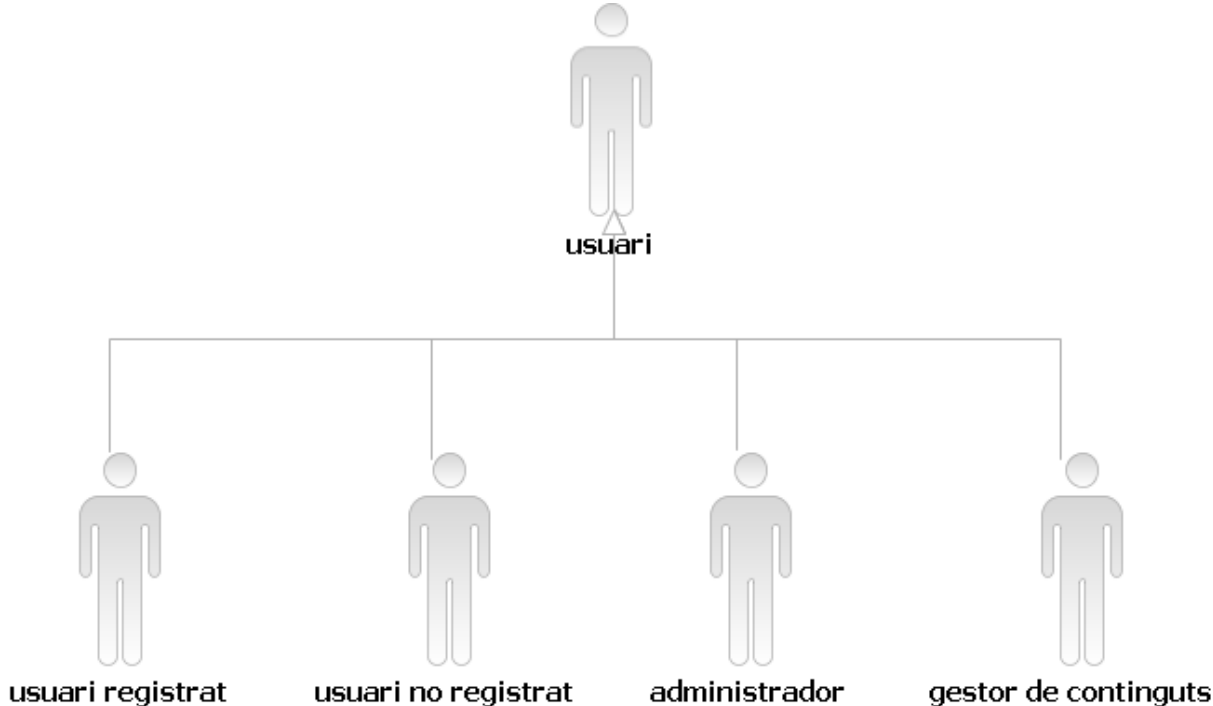
Pel que fa al servidor web, es precisarà un sistema LAMP (Linux, Apache, MySQL, PHP) i per tant un sistema operatiu Linux. Tant la versió de PHP com de MySQL serà mínim la 5.

3 Especificació

En aquest apartat s'explicarà amb precisió mitjançant casos d'ús totes les interaccions que tindran els usuaris amb el sistema. Aquests casos d'ús els extrauré dels requeriments funcionals ja comentats anteriorment. Per cada requisit hi haurà un o més casos d'ús associats. Abans de tot, comentaré els usuaris del sistema:

- **Administrador:** és l'usuari amb més poders, pot gestionar usuaris registrats (eliminar-los o modificar-los) així com gestionar les diferents seccions de que es compona el sistema (Alimentació, Frescs, Basar, etc..) i/o introduir-hi productes. També pot veure estadístiques de la pàgina en general. Només hi ha un administrador en el sistema i aquest no es pot modificar ni crear, ja ve predeterminat amb el sistema.
- **Gestor de continguts:** és la persona encarregada de donar contingut a la web. A diferència de l'administrador, el gestor només pot inserir, modificar i eliminar nous productes mitjançant el panell de control.
- **Usuari no registrat:** és qualsevol persona que visiti la plana però que no pot fer res més que registrar-se en el sistema per a poder generar les seves llistes de la compra.
- **Usuari registrat:** es l'usuari que aprofitarà el sistema, podrà generar llistes de la compra, esborrar-ne, veure comparatives de preus entre supermercats, fer cerques, etc.

A continuació es mostra un esquema *UML* de la jerarquia d'usuaris:



3.1 Diagrama dels casos d'ús

A continuació, es representa gràficament en un diagrama els diferents casos d'ús i els actors que els protagonitzen. Degut a l'extens nombre de cassos d'ús he decidit partir-los en dos diagrames, un primer diagrama amb els casos d'ús de sistema (part d'usuaris registrats) i un altre per al panell de control del sistema.

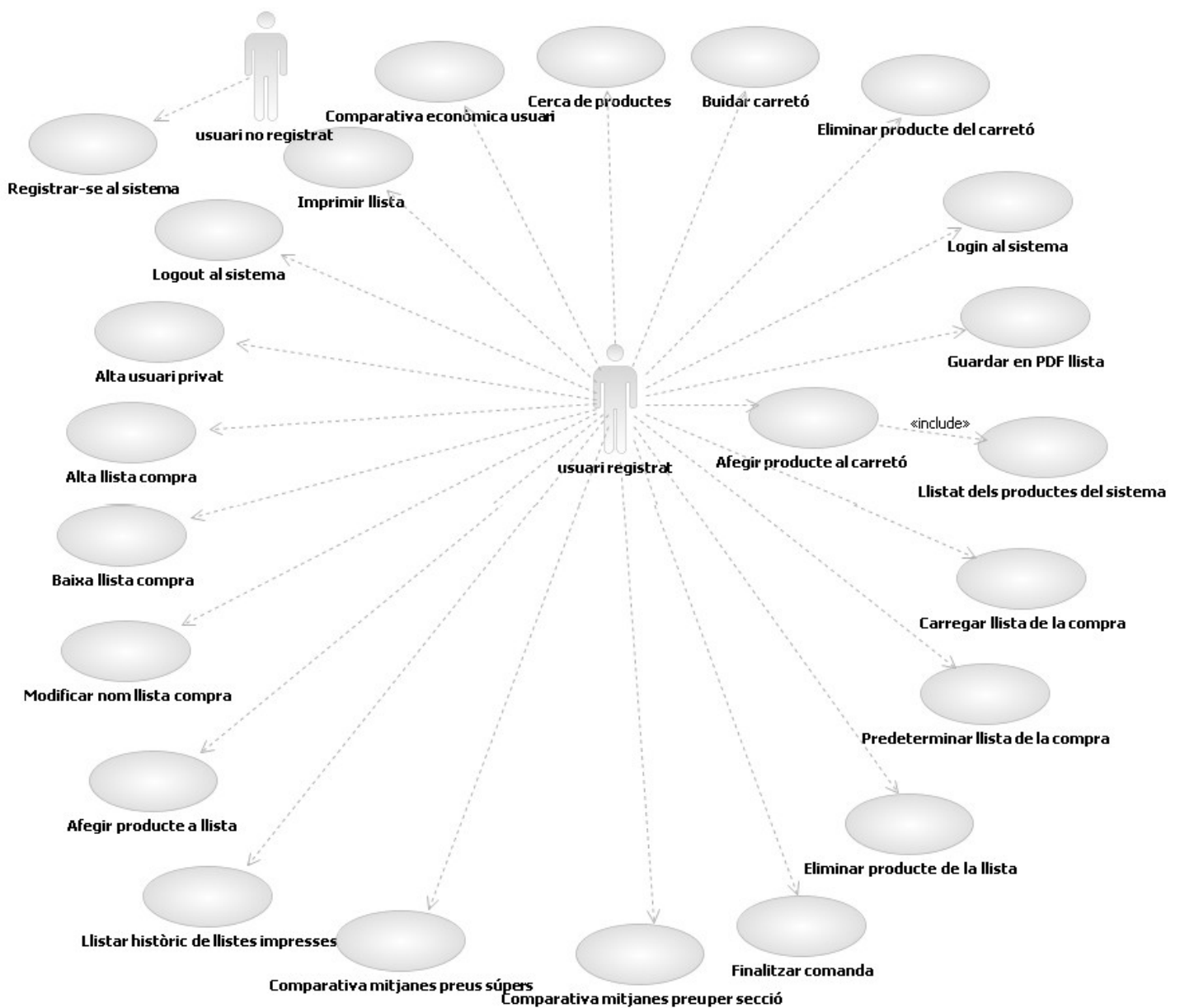
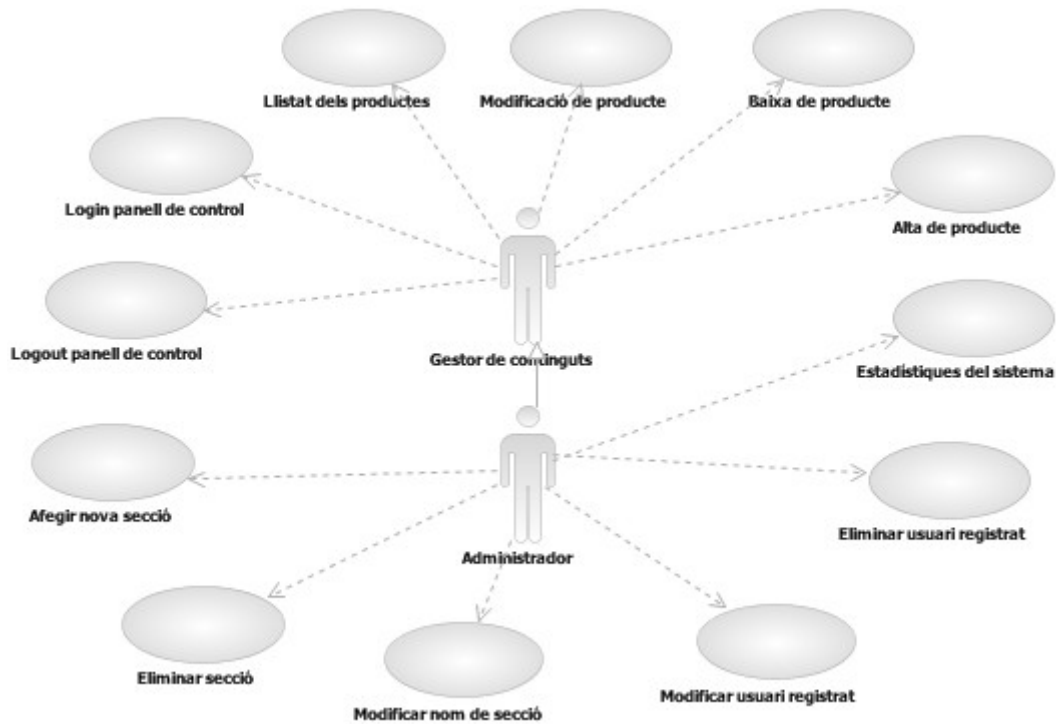


Diagrama de cassos d'ús del panell de control del sistema:



A continuació es definiran textualment els diferents casos d'ús. En ells descriuré la seqüència d'esdeveniments que realitza un actor del sistema per portar a terme un procés que té valor per a ell. Primer de tot identificaré als actors que participen en el cas d'ús anunciat, que pot ser tant una persona física com un sistema hardware, software, etc. Després comentaré quin és el propòsit del cas d'ús i ho acompanyaré amb un petit resum del que fa. Acabaré descrivint els diferents cursos alternatius en que pot desembocar el curs típic d'esdeveniments.

3.2 Registrar-se al sistema

Cas d'ús	Registrar-se al sistema	
Actors	Usuari no registrat	
Propòsit	Donar-se d'alta en el sistema	
Resum	El cas d'ús s'origina quan un usuari no registrat pretén poder utilitzar el sistema	
Curs típic d'esdeveniments	Accions dels actors	Accions del sistema
	<p>1. L'usuari accedeix a la plana principal de l'aplicació i clica l'enllaç per a registrar-se.</p> <p>3. L'usuari omple el formulari i l'envia.</p>	<p>2. El sistema mostra a l'usuari un formulari per omplir el seu nom, la contrasenya, la contrasenya repetida i el gasto mensual actual en la compra.</p> <p>4. El sistema emmagatzema a la base de dades el nou usuari i li confirma que el registre s'ha efectuat correctament.</p>
Cursos alternatius	4. Si la contrasenya i la contrasenya repetida no encaixen o si la contrasenya de l'usuari no supera un cert llindar de seguretat el sistema mostra una alerta d'error a l'usuari i no es procedeix al registre. Així mateix si el nom d'usuari ja existeix s'avisarà a l'usuari no registrat de que canviï el nom.	

3.3 Alta d'usuari privat

Cas d'ús	Alta d'usuari privat	
Actors	Administrador	
Propòsit	Donar d'alta un usuari privat	
Resum	Aquest cas d'ús origina un nou usuari privat per al panell de control, aquest tindrà poders de poder modificar el contingut del sistema.	
Curs típic d'esdeveniments	Accions dels actors	Accions del sistema
	<p>1. L'administrador pretén donar d'alta un nou gestor de continguts.</p> <p>3. L'administrador omple el formulari i l'envia.</p>	<p>2. El sistema presenta un formulari on inserir un nom pel gestor, contrasenya i la contrasenya repetida.</p> <p>4. El sistema emmagatzema a la base de dades el nou gestor de continguts i confirma que el registre s'ha efectuat correctament.</p>
Cursos alternatius	4. Si la contrasenya i la contrasenya repetida no encaixen o si la contrasenya de l'usuari no supera un cert llindar de seguretat el sistema mostra una alerta d'error a l'usuari i no es procedeix al registre. Així mateix si el nom pel gestor ja existeix s'avisarà a l'administrador de que canviï el nom.	

3.4 Alta llista de la compra

Cas d'ús	Alta llista de la compra	
Actors	Usuari registrat	
Propòsit	Crear una nova llista de la compra	
Resum	Un usuari registrat desitja crear una nova llista de la compra per afegir-hi nous productes.	
Curs típic d'esdeveniments	Accions dels actors	Accions del sistema
	<p>1. L'usuari registrat clica en l'apartat "Llistes" del menú del sistema.</p> <p>3. L'usuari clica l'icona Crear nova llista.</p> <p>5. L'usuari prem enter per finalitzar.</p>	<p>2. El sistema obre una nova finestra llistant totes les llistes de la compra actuals.</p> <p>4. El sistema mostra un camp de text on escriure el nom de la nova llista.</p> <p>6. El sistema emmagatzema a la base de dades la nova llista.</p>
Cursos alternatius	6. Si l'usuari ha premut enter o tabulador sense haver escrit cap nom el sistema l'alertarà de que no pot continuar.	

3.5 Eliminar llista de la compra

Cas d'ús	Eliminar llista de la compra	
Actors	Usuari registrat	
Propòsit	Eliminar una llista de la compra	
Resum	Aquest cas d'ús elimina una la llista de la compra relacionada amb l'usuari registrat que l'efectua.	
Curs típic d'esdeveniments	Accions dels actors	Accions del sistema
	<p>1. L'usuari registrat clica en l'apartat "Llistes" del menú del sistema.</p> <p>3. L'usuari clica l'icona amb una creu vermella per tal d'eliminar la llista.</p>	<p>2. El sistema obre una nova finestra llistant totes les llistes de la compra actuals.</p> <p>4. El sistema demana confirmació abans d'eliminar la llista. Si la confirmació es positiva s'eliminarà la llista i tots les associacions dels productes relacionats amb ella.</p>
Cursos alternatius	4. L'usuari registrat no accepta la confirmació, es torna al pas 2 d'aquest cas d'ús.	

3.6 Modificar nom de la llista de la compra

Cas d'ús	Modificar nom de la llista de la compra	
Actors	Usuari registrat	
Propòsit	Modificar el nom d'una llista.	
Resum	L'actor del cas d'ús desitja modificar el nom d'una llista.	
Curs típic d'esdeveniments	Accions dels actors	Accions del sistema
	<p>1. L'actor clica sobre la pestanya "Llistes" del menú del sistema.</p> <p>3. L'usuari clica l'icona associada a la llista que vol modificar-li el nom.</p> <p>5. L'usuari modifica el nom i clica la tecla enter per guardar els canvis.</p>	<p>2. El sistema obre una nova finestra mostrant les llistes de que disposa actualment l'usuari connectat.</p> <p>4. El nom de la llista es torna una caixa de text modificable tot mostrant l'actual nom de la llista.</p> <p>6. El sistema emmagatzema els canvis i torna al pas 2 però mostrant la llista actualitzada.</p>
Cursos alternatius	5. L'usuari no escriu cap valor en el camp de text, el sistema l'avisa de que entri un nom per a poder continuar.	

3.7 Afegir producte a la llista de la compra

Cas d'ús	Afegir producte a la llista de la compra	
Actors	Usuari registrat	
Propòsit	Afegir un nou producte a la llista de la compra actual	
Resum	L'usuari registrat desitja afegir un nou producte a l'actual llista de la compra.	
Curs típic d'esdeveniments	Accions dels actors	Accions del sistema
	<p>1. El visitant clica sobre la secció del sistema on hi hagi el producte que vol treure de la seva llista.</p> <p>3. El visitant selecciona la casella de verificació del producte que li interessi afegir.</p>	<p>2. El sistema mostra per cada producte de la secció una casella de verificació. Si la casella no està seleccionada significa que el producte no es troba a la llista.</p> <p>4. El sistema demana confirmació de si vol afegir aquell producte a la actual llista. Si l'usuari accepta s'emmagatzema aquest canvi a la base de dades.</p>
Cursos alternatius	4. L'usuari no accepta i es torna al pas 2.	

3.8 Eliminar producte de la llista de la compra

Cas d'ús	Eliminar producte de la llista de la compra	
Actors	Usuari registrat	
Propòsit	Eliminar un producte de la llista de la compra actual.	
Resum	L'usuari registrat desitja eliminar un producte de l'actual llista de la compra.	
Curs típic d'esdeveniments	Accions dels actors	Accions del sistema
	<p>1. El visitant clica sobre la secció del sistema on hi hagi el producte que vol treure de la seva llista.</p> <p>3. El visitant des-selecciona el producte que li interessa donar de baixa de la llista.</p>	<p>2. El sistema mostra per cada producte de la secció una casella de verificació. Si està seleccionada significa que el producte es troba a la llista.</p> <p>4. El sistema elimina de la actual llista el producte clicat tot demanat confirmació a l'usuari abans de fer-ho.</p>
Cursos alternatius	4. L'usuari registrat no accepta la confirmació, es torna al pas 2 d'aquest cas d'ús.	

3.9 Predeterminar llista de la compra

Cas d'ús	Predeterminar llista de la compra	
Actors	Usuari registrat	
Propòsit	Definir com a llista actual de la compra un altre llista.	
Resum	L'actual usuari registrat pot disposar de diverses llistes de la compra (p. ex: una llista de la compra per entre setmana, un altre per a dies especials com Nadal, Cap d'any, etc..). L'usuari pot canviar de llista quan vulgui amb aquest cas d'ús.	
Curs típic d'esdeveniments	Accions dels actors	Accions del sistema
	<p>1. L'usuari pretén canviar la actual llista, per a això clica sobre la pestanya "Llistes" del menú del sistema.</p> <p>3. D'entre les diferents llistes l'usuari clica l'icona de predeterminar llista.</p>	<p>2. El sistema obra una nova finestra llistant les diferents llistes de que disposa l'usuari. La actual llista predeterminada es reconeixerà per estar en cursiva.</p> <p>4. El sistema emmagatzema demana confirmació dels canvis. Si l'usuari confirma, a partir d'ara tot producte entrat a la llista s'associarà a aquesta llista de la compra. Es torna al pas 2.</p>
Cursos alternatius	4. Si l'usuari no accepta es torna al pas 2.	

3.10 Carregar llista de la compra al carretó

Cas d'ús	Carregar llista de la compra al carretó	
Actors	Usuari registrat	
Propòsit	Carregar una llista de la compra	
Resum	L'usuari pretén carregar al carretó de la compra una llista ja creada anteriorment.	
Curs típic d'esdeveniments	Accions dels actors	Accions del sistema
	<ol style="list-style-type: none"> 1. L'usuari registrat clica en l'apartat "Llistes" del menú del sistema. 3. L'usuari clica l'icona d'una carpeta groga oberta amb l'objectiu de carregar la llista. 	<ol style="list-style-type: none"> 2. El sistema obre una nova finestra llistant totes les llistes de la compra actuals. 4. El sistema buida l'actual carretó i seguidament hi carrega els productes relacionats amb la llista clicada. A continuació es tanca la finestra oberta.
Cursos alternatius	4. Si la llista no conté cap producte associat avisarà a l'usuari que aquesta operació no es pot realitzar.	

3.11 Llistat dels productes del sistema

Cas d'ús	Llistat dels productes del sistema	
Actors	Usuari registrat	
Propòsit	Llistar els productes del sistema	
Resum	Es mostra un llistat per cada secció amb els productes donats d'alta en el sistema tot assenyalant el preu del supermercat més baix però amb opció de seleccionar un altre supermercat (per comoditat o preferència per exemple).	
Curs típic d'esdeveniments	Accions dels actors	Accions del sistema
	1. L'usuari pretén veure els productes que hi ha en una secció qualsevol A.	2. El sistema rep la petició i mostra en la plana els productes de la secció A ordenats alfabèticament tot mostrant en cada producte el preu més baix dels entrats dels 3 supermercats. També per cada producte mostra un selector per afegir-lo o retirar-lo de l'actual carretó així com d'un quadre seleccionable amb l'opció d'afegir-lo a la llista habitual de la compra.
Cursos alternatius	2. Pot ocórrer que un producte no tingui els 3 preus entrats en el supermercat i sols en tingui 1 o 2. En tal cas, mostrarà igualment el preu més baix.	

3.12 Afegir producte al carretó

Cas d'ús	Afegir producte al carretó	
Actors	Usuari registrat	
Propòsit	L'usuari vol afegir un producte al carretó.	
Resum	Per tal d'afegir un producte al carretó de la llista, l'usuari deurà indicar una quantitat d'aquest.	
Curs típic d'esdeveniments	Accions dels actors	Accions del sistema
	<p>1. L'usuari selecciona una secció del sistema.</p> <p>3. L'usuari disposa per cada producte de dos selectors, un per augmentar i un altre per disminuir una unitat d'un producte en concret. Si augmenta una unitat s'afegirà automàticament en el carretó, mentre siguin positives les unitats s'aniran incrementant en el carretó. Tanmateix pot seleccionar el super que li interessi.</p>	<p>2. El sistema mostra un llistat dels productes entrats en aquella secció.</p> <p>4. El sistema emmagatzema en la cistella de la compra l'alta del nou producte junt amb les seves unitats i super i ho mostra en el panell de carretó.</p>
Cursos alternatius	No n'hi ha	

3.13 Eliminar producte del carretó

Cas d'ús	Eliminar producte del carretó	
Actors	Usuari registrat	
Propòsit	Eliminar un producte del carretó	
Resum	L'usuari s'ho repensa i desitja eliminar un producte ja inserit en el carretó.	
Curs típic d'esdeveniments	Accions dels actors	Accions del sistema
	<p>1. L'usuari desitja eliminar un producte del carretó.</p> <p>3. L'usuari disposa per cada producte (de la taula del carretó) d'una icona amb forma de cubell de brossa per tal d'eliminar aquell producte en concret. Clica el que no li interressi mantenir.</p> <p>5. L'usuari confirma o denega la confirmació.</p>	<p>2. El sistema mostra en tot moment un a taula amb el conjunt dels productes actualment inserits en el carretó.</p> <p>4. El sistema demana confirmació per eliminar el producte del carretó.</p> <p>6. Si l'usuari confirma afirmativament aleshores s'elimina del carretó el producte i es mostra de nou el carretó actualitzat.</p>
Cursos alternatius	<p>3. Un altre forma d'eliminar un producte en concret es accedint a la seva secció i seleccionant-li 0 unitats en comptes de les actualment entrades al carretó.</p> <p>6. Si l'usuari confirma negativament no ocorre res i es segueix llistant els productes tant del carretó com de la secció.</p>	

3.14 Buidar carretó

Cas d'ús	Buidar carretó	
Actors	Usuari registrat	
Propòsit	Buidar el carretó	
Resum	L'usuari es desdiu de la llista o en vol fer una nova des de zero i decideix buidar tot el carretó.	
Curs típic d'esdeveniments	Accions dels actors	Accions del sistema
	<p>1. L'usuari decideix buidar tot el carretó. Clica en l'enllaç de "Buidar carretó" situat a sota de la taula dels productes actualment inserits en el carretó.</p> <p>3. L'usuari confirma o no l'alerta del sistema.</p>	<p>2. El sistema demana confirmació per eliminar tot el carretó.</p> <p>4. Si es confirma positivament, el sistema buida el carretó, el mostra buit i es torna a la plana d'inici del sistema.</p>
Cursos alternatius	4. Si es confirma negativament, no passa res, no es borra el carretó i l'usuari segueix en la plana que estava.	

3.15 Finalitzar comanda

Cas d'ús	Finalitzar comanda	
Actors	Usuari registrat	
Propòsit	Finalitzar una comanda	
Resum	L'usuari creu que ja te entrats al carretó tots els productes que realment precisa per a la llista i desitja finalitzar per a poder imprimir la llista en paper.	
Curs típic d'esdeveniments	Accions dels actors	Accions del sistema
	<p>1. L'usuari decideix finalitzar i clica en l'enllaç "Finalitzar comanda" que disposa sota de la taula del carretó.</p> <p>3. L'usuari confirma o no l'alerta del sistema.</p>	<p>2. El sistema pregunta a l'usuari si realment desitja finalitzar.</p> <p>4. Si es confirma positivament, el sistema finalitza la llista, grava els canvis en l'historial de compra del client, buida el carretó i mostra a l'usuari la llista de la compra finalitzada amb opció d'imprimir o guardar en PDF.</p>
Cursos alternatius	4. Si es confirma negativament, no ocorre res, no es finalitza la llista i l'usuari segueix en la plana que estava.	

3.16 Llistar històric de llistes impreses

Cas d'ús	Llistar històric de llistes impreses	
Actors	Usuari registrat	
Propòsit	Llistar els històrics de les llistes	
Resum	L'usuari registrat vol veure quines llistes de la compra ha ja imprès (o finalitzat) temps enrere. Es descriurà en detall què es va llistar, el super i la quantitat.	
Curs típic d'esdeveniments	Accions dels actors	Accions del sistema
	<p>1. L'usuari registrat clica en l'apartat "Historial" del menú del sistema.</p> <p>3. L'usuari pot clicar sobre l'icona d'un ull per a visualitzar en detall l'historial.</p>	<p>2. El sistema obre una nova finestra llistant els historials de les llistes ja impreses (finalitzades). En concret es mostra el dia en que es va crear, la quantitat total d'articles de la llista i el preu total que suposaria.</p> <p>4. El sistema desplega en detall l'historial demanat per l'usuari.</p>
Cursos alternatius	No n'hi ha.	

3.17 Comparativa de les mitjanes de preus dels supermercats

Cas d'ús	Comparativa de les mitjanes de preus dels supermercats	
Actors	Usuari registrat	
Propòsit	Veure les mitjanes globals de preus dels 3 supermercats	
Resum	Es mostrarà amb un gràfic animat les mitjanes globals dels 3 supermercats existents en el sistema. Amb aquest gràfic es vol donar a conèixer quin supermercat es globalment més barat.	
Curs típic d'esdeveniments	Accions dels actors	Accions del sistema
	1. L'usuari registrat clica en l'apartat "Comparativa" del menú del sistema.	2. El sistema obre una nova finestra mostrant un gràfic amb les mitjanes globals dels 3 súpers.
Cursos alternatius	No n'hi ha	

3.18 Comparativa de les mitjanes de preus per secció

Cas d'ús	Comparativa de les mitjanes de preus per secció	
Actors	Usuari registrat	
Propòsit	Visualitzar comparatives de les mitjanes dels preus dels supermercats per secció	
Resum	L'usuari decideix veure per cada secció dels 3 supermercats, en quin es més barat comprar una tipologia de producte o altre.	
Curs típic d'esdeveniments	Accions dels actors	Accions del sistema
	<p>1. L'usuari registrat clica en l'apartat "Comparativa" del menú del sistema.</p> <p>3. L'usuari registrat clica en l'enllaç de Comparativa per seccions.</p>	<p>2. El sistema obre una nova finestra mostrant per defecte el gràfic de barres de les mitjanes globals dels 3 súpers.</p> <p>4. El sistema mostra un gràfic de barres amb les mitjanes de preus per secció dels 3 supermercats.</p>
Cursos alternatius	No n'hi ha	

3.19 Comparativa de la trajectòria econòmica de l'usuari

Cas d'ús	Comparativa de la trajectòria econòmica de l'usuari	
Actors	Usuari registrat	
Propòsit	Veure una comparativa de la trajectòria econòmica de l'usuari	
Resum	Si l'usuari ja ha finalitzat una o més llistes de la compra amb el sistema, es mostrarà un gràfic lineal del que l'usuari gastava abans d'utilitzar el sistema i després.	
Curs típic d'esdeveniments	Accions dels actors	Accions del sistema
	<p>1. L'usuari registrat clica en l'apartat "Comparativa" del menú del sistema.</p> <p>3. L'usuari registrat clica en l'enllaç de Comparativa historial client.</p>	<p>2. El sistema obre una nova finestra mostrant un gràfic de barres amb les mitjanes globals dels 3 súpers.</p> <p>4. El sistema mostra una gràfica lineal amb el que l'usuari va indicar (gasto mensual) el primer dia al registrar-se fins al dia d'avui després d'utilitzar el sistema.</p>
Cursos alternatius	No n'hi ha	

3.20 Cerca de productes

Cas d'ús	Cerca de productes	
Actors	Usuari registrat	
Propòsit	Realitzar una cerca concreta de productes	
Resum	L'usuari no troba un producte o prefereix cercar-lo ràpidament i decideix realitzar una cerca més específica.	
Curs típic d'esdeveniments	Accions dels actors	Accions del sistema
	<p>1. L'usuari decideix realitzar una cerca. Selecciona l'apartat Cercador del menú del sistema.</p> <p>3. L'usuari introdueix el nom d'un producte en concret o una característica del mateix.</p> <p>5. L'usuari selecciona del desplegable que presenta el sistema un producte en concret o clica enter ignorant el desplegable.</p>	<p>2. El sistema obra una nova finestra amb un requadre per inserir el text a cercar.</p> <p>4. El sistema mostra a mesura que l'usuari escriu el nom o característica a cercar, un desplegable amb ocurrències que encaixen amb lo cercat.</p> <p>6. El sistema tanca la finestra oberta en el pas 2 i presenta el/els resultat/s en la finestra de productes.</p>
Cursos alternatius	No n'hi ha	

3.21 Imprimir llista

Cas d'ús	Imprimir llista	
Actors	Usuari registrat	
Propòsit	L'usuari decideix imprimir una llista.	
Resum	Al finalitzar la comanda es presenta l'opció d'imprimir la llista de la compra resultant. Aquesta mateixa opció es pot trobar en la secció de "Llistes".	
Curs típic d'esdeveniments	Accions dels actors	Accions del sistema
	<p>1. L'usuari decideix imprimir la llista i finalitza la comanda o selecciona l'opció imprimir de la secció "llistes".</p> <p>3. L'usuari selecciona l'opció imprimir.</p>	<p>2. El sistema obra una finestra amb la llista, mostrant com quedarà al imprimir-se.</p> <p>4. El sistema envia a la impressora la llista a imprimir.</p>
Cursos alternatius	3. L'usuari decideix no imprimir, tanca la finestra i s'acaba el cas d'ús tot tornant a la plana de productes.	

3.22 Guardar en PDF una llista

Cas d'ús	Guardar en PDF una llista	
Actors	Usuari registrat	
Propòsit	L'usuari decideix guardar en un arxiu una llista.	
Resum	Al finalitzar la comanda es presenta l'opció de guardar la llista de la compra resultant. Aquesta mateixa opció es pot trobar en la secció de "Listes".	
Curs típic d'esdeveniments	Accions dels actors	Accions del sistema
	<p>1. L'usuari decideix guardar la actual llista o una de ja feta anteriorment.</p> <p>3. L'usuari selecciona l'opció de guardar llista en PDF.</p>	<p>2. El sistema obra una finestra amb la llista, mostrant el seu contingut.</p> <p>4. El sistema obre un diàleg per guardar o obrir la llista seleccionada.</p>
Cursos alternatius	3. L'usuari decideix no guardar en PDF la llista, tanca la finestra i s'acaba el cas d'ús tot tornat a la plana de productes.	

3.23 Accés (“login”) al sistema

Cas d'ús	Accés (“login”) al sistema	
Actors	Usuari registrat	
Propòsit	Que l'usuari s'identifiqui al sistema per tal d'accedir a les seves funcionalitats	
Resum	Cal que tots els usuaris s'identifiquin al sistema per tal de poder accedir a les seves funcionalitats amb les seva configuració pròpia.	
Curs típic d'esdeveniments	Accions dels actors	Accions del sistema
	1. L'usuari introdueix el seu nom d'usuari i contrasenya.	2. El sistema valida la informació i permet l'accés a l'usuari.
Cursos alternatius	2. La informació és errònia i el sistema no permet l'accés a l'usuari. Se li ho notifica a l'usuari.	

3.24 Sortida (“logout”) del sistema

Cas d'ús	Sortida (“logout”) del sistema	
Actors	Usuari registrat	
Propòsit	L'usuari vol tancar la seva sessió i deixar de tenir accés al sistema	
Resum	Per tal d'indicar que deixa d'utilitzar el sistema i tancar la sessió, l'usuari fa logout.	
Curs típic d'esdeveniments	Accions dels actors	Accions del sistema
	<ol style="list-style-type: none"> 1. L'usuari indica al sistema la intenció de tancar la sessió. 3. L'usuari accepta. 	<ol style="list-style-type: none"> 2. El sistema demana confirmació. 4. El sistema tanca la sessió i no permet l'accés a les seves funcionalitats.
Cursos alternatius	3. L'usuari denega i continua la sessió.	

3.25 Entrada (“login”) al panell de control del sistema

Cas d'ús	Entrada (“login”) al panell de control del sistema	
Actors	Gestor de continguts, Administrador	
Propòsit	Que l'administrador o el gestor de continguts s'identifiquin al sistema per tal d'accedir al panell de control	
Resum	Per tal d'accedir a la zona de control del sistema, cal que els usuaris amb tals permisos s'identifiquin.	
Curs típic d'esdeveniments	Accions dels actors	Accions del sistema
	1. L'administrador o gestor de continguts introdueix el seu nom d'usuari, contrasenya i les lletres de verificació anti-bot's.	2. El sistema valida la informació i permet l'accés a l'usuari.
Cursos alternatius	2. La informació és errònia i el sistema no permet l'accés a l'usuari. Se li ho notifica a l'usuari. Per raons de seguretat, si l'usuari falla tres cops aquest pas, se li denega l'entrada durant uns minuts.	

3.26 Sortida (“logout”) del panell de control del sistema

Cas d'ús	Sortida (“Logout”) del panell de control del sistema	
Actors	Gestor de continguts, Administrador	
Propòsit	L'usuari vol tancar la seva sessió i deixar de tenir accés al panell de control del sistema.	
Resum	Per tal d'indicar que deixa d'utilitzar el panell de control, l'usuari fa logout.	
Curs típic d'esdeveniments	Accions dels actors	Accions del sistema
	<ol style="list-style-type: none"> 1. L'usuari indica al sistema la intenció de tancar la sessió. 3. L'usuari accepta. 	<ol style="list-style-type: none"> 2. El sistema demana confirmació. 4. El sistema tanca la sessió i no permet l'accés a les seves funcionalitats.
Cursos alternatius	3. L'usuari denega i continua la sessió.	

3.27 Afegir nova secció al sistema

Cas d'ús	Afegir nova secció al sistema	
Actors	Administrador	
Propòsit	Afegir una nova secció de productes	
Resum	L'administrador decideix d'afegir una nova secció	
Curs típic d'esdeveniments	Accions dels actors	Accions del sistema
	<p>1. L'administrador decideix donar d'alta una nova secció on posar-hi productes.</p> <p>3. L'usuari pot afegir una nova secció filla (p. ex: begudes isotòniques, filla de Begudes) d'un altre ja existent o afegir una nova secció "pare" (p. ex: carn). Insereix un nom nou i envia el formulari tot prement enter.</p>	<p>2. El sistema mostra un formulari amb l'opció d'afegir una nova secció. Alhora mostra un desplegable amb un llistat de les actuals seccions.</p> <p>4. El sistema valida el nom i l'emmagatzema en el sistema.</p>
Cursos alternatius	4. Si ocorre un error alhora d'emmagatzemar el nom, se li ho notifica a l'usuari i no es guarden els canvis.	

3.28 Eliminar secció del sistema

Cas d'ús	Eliminar secció del sistema	
Actors	Administrador	
Propòsit	Eliminar secció de productes	
Resum	L'usuari decideix que una secció de productes ja no es funcional o no li interessa mantenir-la activa.	
Curs típic d'esdeveniments	Accions dels actors	Accions del sistema
	<p>1. L'administrador decideix esborrar una secció de productes.</p> <p>3. L'usuari clica en l'icona d'eliminar.</p> <p>5. L'usuari accepta.</p>	<p>2. El sistema mostra un llistat amb totes les seccions i subseccions que te emmagatzemades.</p> <p>4. El sistema demana confirmació per a donar de baixa la secció i les seves subseccions si aquesta es una secció "pare".</p> <p>4. El sistema dona de baixa la secció junt amb les seves subseccions si es una secció pare. Tanmateix elimina els productes associats a la secció o subsecció.</p>
Cursos alternatius	5. L'usuari no accepta,aleshores es torna al pas 2.	

3.29 Modificar el nom d'una secció del sistema

Cas d'ús	Modificar el nom d'una secció del sistema	
Actors	Administrador	
Propòsit	Modificar el nom d'una secció	
Resum	L'administrador vol reanomenar una secció	
Curs típic d'esdeveniments	Accions dels actors	Accions del sistema
	<p>1. L'administrador té l'intenció de reanomenar una secció.</p> <p>3. L'administrador clica en el nom de la secció que pretén canviar de nom.</p> <p>5. L'administrador modifica el nom i envia el formulari.</p>	<p>2. El sistema mostra un llistat amb totes les seccions i subseccions de que disposa el sistema.</p> <p>4. El sistema obre una nova finestra amb un formulari on apareix un camp de text amb el nom de la secció actual i un botó per enviar canvis.</p> <p>6. La finestra oberta es tanca, s'actualitza el sistema amb la nova informació i es mostra de nou el llistat però actualitzat.</p>
Cursos alternatius	5. L'administrador decideix no modificar el nom, tanca la finestra oberta i es torna al pas 2.	

3.30 Modificar usuari registrat

Cas d'ús	Modificar usuari registrat	
Actors	Administrador	
Propòsit	Modificar usuaris del sistema.	
Resum	Modificar els usuaris registrats del sistema. Es podrà modificar el nom d'usuari o la contrasenya d'un usuari registrat.	
Curs típic d'esdeveniments	Accions dels actors	Accions del sistema
	<p>1. L'administrador del sistema decideix modificar un usuari.</p> <p>3. L'administrador indica qui és l'usuari a modificar.</p> <p>5. L'administrador modifica el formulari (nom d'usuari i/o contrasenya).</p>	<p>2. El sistema mostra un llistat de tots els usuaris registrats en el sistema.</p> <p>4. El sistema obra una nova finestra tot mostrant un formulari amb l'actual informació de l'usuari.</p> <p>6. El sistema valida les dades introduïdes, guarda els canvis si tot es correcte i torna al pas 2.</p>
Cursos alternatius	6. Si el nou nom d'usuari inserit ja existeix en el sistema s'avisarà a l'administrador de l'error i no es continuarà ni es salvaran els canvis. Tanmateix si la nova contrasenya i la repetida no encaixen no es prosseguirà o si la contrasenya no es considera forta (és a dir, no arriba a un cert llindar).	

3.31 Eliminar usuari registrat

Cas d'ús	Eliminar usuari registrat	
Actors	Administrador	
Propòsit	Eliminar un usuari registrat	
Resum	Un usuari registrat decideix donar-se de baixa del sistema, li ho notifica a l'administrador.	
Curs típic d'esdeveniments	Accions dels actors	Accions del sistema
	<p>1. L'administrador decideix donar de baixa un usuari registrat i accedeix a l'apartat Eliminar usuaris registrats en el panell de control.</p> <p>3. L'administrador clica l'icona d'eliminar d'un usuari en concret.</p> <p>5. L'administrador confirma.</p>	<p>2. El sistema mostra un llistat amb els usuaris registrats actualment i amb un icona per eliminar-los.</p> <p>4. El sistema demana confirmació per a donar de baixa l'usuari seleccionat.</p> <p>6. El sistema dona de baixa l'usuari en concret així com les seves llistes de la compra, històrics i associacions que hi hagi en la base de dades.</p>
Cursos alternatius	5. L'administrador no confirma, es torna al pas 2.	

3.32 Estadístiques del sistema

Cas d'ús	Estadístiques del sistema	
Actors	Administrador	
Propòsit	L'administrador decideix veure les estadístiques del sistema.	
Resum	L'administrador vol veure quins apartats son els més utilitzats per els usuaris registrats.	
Curs típic d'esdeveniments	Accions dels actors	Accions del sistema
	1. L'administrador decideix veure les estadístiques del sistema. En el panell de control clica en estadístiques en la plana principal.	2. El sistema mostra un gràfic rodó amb el nombre total de clics rebuts (en tant per cent) per part dels usuaris registrats a cada apartat del sistema i una taula lateral amb els clics rebuts (sense tant per cent).
Cursos alternatius	No n'hi ha	

3.33 Alta d'un nou producte al sistema

Cas d'ús	Alta d'un nou producte al sistema	
Actors	Gestor de continguts, Administrador	
Propòsit	Donar d'alta un nou producte	
Resum	Es decideix donar d'alta un nou producte en una secció concreta del sistema.	
Curs típic d'esdeveniments	Accions dels actors	Accions del sistema
	<p>1. Es pretén afegir un nou producte. Un dels actors del cas d'ús clica en "Afegir producte" en el panell de control.</p> <p>3. L'actor omple el formulari i l'envia.</p>	<p>2. El sistema mostra un formulari a omplir amb el nom del producte, el preu del super A, del B i del C, un desplegable amb les diferents seccions on poder fixar el producte, una fotografia i unes paraules clau que l'identifiquin.</p> <p>4. El sistema valida les dades i emmagatzema els canvis en el sistema, es torna a mostrar el formulari de l'apartat 2 buit.</p>
Cursos alternatius	4. El sistema no valida correctament les dades, ho notifica a l'usuari i es torna al pas 2.	

3.34 Baixa d'un producte del sistema

Cas d'ús	Baixa d'un producte del sistema	
Actors	Gestor de continguts, Administrador	
Propòsit	Donar de baixa un producte	
Resum	Es decideix donar de baixa un producte del sistema	
Curs típic d'esdeveniments	Accions dels actors	Accions del sistema
	<p>1. Es pretén donar de baixa un producte. Un dels actors del cas d'ús clica en "Eliminar producte" en el panell de control.</p> <p>3. L'actor clica l'icona d'eliminar sobre el producte que desitja eliminar.</p> <p>5. L'actor confirma.</p>	<p>2. El sistema llista tots els productes ordenats alfabèticament.</p> <p>4. El sistema demana confirmació per eliminar el producte.</p> <p>6. S'elimina el producte del sistema, s'emmagatzema els canvis i es torna al punt 2 amb la llista actualitzada.</p>
Cursos alternatius	5. L'actor no confirma, es torna al pas 2.	

3.35 Modificació d'un producte del sistema

Cas d'ús	Modificació d'un producte del sistema	
Actors	Gestor de continguts, Administrador	
Propòsit	Modificar un producte	
Resum	Es decideix modificar un producte ja donat d'alta en el sistema.	
Curs típic d'esdeveniments	Accions dels actors	Accions del sistema
	<p>1. Es pretén modificar un producte. Un dels actors del cas d'ús clica en "Modificar producte" en el panell de control.</p> <p>3. L'actor selecciona un producte a modificar de la llista.</p> <p>5. L'actor modifica el formulari i l'envia.</p>	<p>2. El sistema mostra un llistat amb tots els productes emmagatzemats al sistema.</p> <p>4. El sistema mostra un formulari amb els valors actuals associats al producte i amb un botó d'enviar.</p> <p>6. El sistema valida les dades i emmagatzema els canvis en el sistema, es torna a mostrar el formulari de l'apartat 2 amb els valors modificats.</p>
Cursos alternatius	6. El sistema no valida correctament les dades, ho notifica a l'usuari i es torna al pas 2.	

3.36 Llistat dels productes

Cas d'ús	Llistat dels productes	
Actors	Gestor de continguts, Administrador	
Propòsit	Llistat dels productes	
Resum	Alhora de modificar o eliminar un producte es mostra prèviament un llistat amb els productes del sistema.	
Curs típic d'esdeveniments	Accions dels actors	Accions del sistema
	1. L'actor pretén donar de baixa o eliminar un producte.	2. El sistema mostra un llistat amb la totalitat de productes ordenats alfabèticament. Si el llistat es massa gran es paginaran els resultats.
Cursos alternatius	No n'hi ha	

4 Avaluació tecnològica

En aquest capítol, amb l'objectiu de facilitar la comprensió de la memòria, s'intentarà explicar amb claredat les tecnologies que s'utilitzaran en la realització del projecte.

Tanmateix s'explicarà perquè s'ha cregut oportú utilitzar aquestes tecnologies i no altres.

Per tal de mostrar la plana s'utilitzarà un servidor web Apache, PHP 5 per a crear les pàgines dinàmiques, i Javascript per complir amb els requisits no funcionals de seguretat. Tot i així, per reforçar la seguretat, s'utilitzarà també PHP per comprovar els paràmetres rebuts per l'usuari, ja que Javascript pot ser deshabilitat per qualsevol en el seu navegador.

A continuació s'expliquen totes les tecnologies aplicades durant el desenvolupament del projecte i el seu motiu.

4.1 Arquitectura client/servidor

Donat que el sistema s'utilitza en un entorn web, s'ha cregut convenient explicar una mica com funciona l'arquitectura client/servidor.

L'arquitectura client/servidor consisteix bàsicament en un programa client que realitza peticions a un altre programa (el servidor, en el nostre cas, Apache) que li dóna resposta.

En un model com aquest, les aplicacions es divideixen de manera que en el servidor hi trobem la part a ser compartida per varis usuaris, i en el client sols hi roman la part particular de cada usuari.

Els clients interactuen amb el servidor sovint de manera gràfica. Un servidor proporciona un o varis serveis al client, aquest realitza una petició i el servidor li retorna una resposta.

Normalment, el servidor es troba situat en una ubicació diferent a on es troba l'usuari. Contràriament, en el cas d'aquest projecte, el servidor i el client es troben en el mateix ordenador.

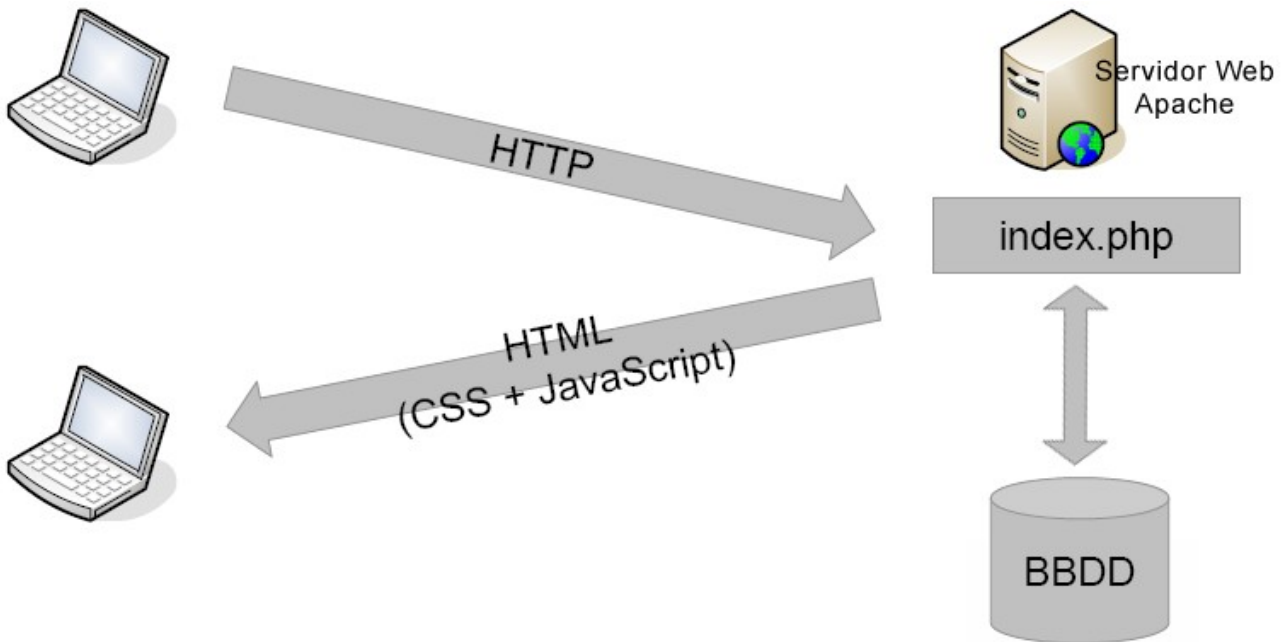
Per tal que els clients i els servidors puguin comunicar-se, es requereix un mecanisme de comunicació que proporcioni les funcions de direccionament i transport. En el nostre cas, aquest mecanisme és Internet.

A grosso modo, l'arquitectura client/servidor es podria resumir en aquestes característiques:

- El servidor presenta a tots els seus clients una interfície única i ben definida.
- El client no necessita conèixer la lògica del servidor, només la seva interfície externa.
- El client no depèn de la ubicació física del servidor, ni del tipus d'equip físic en el qual es troba, ni del seu sistema operatiu.

En el nostre cas, el client serà un navegador web que s'encarregarà de presentar informació i realitzar peticions (mitjançant HTTP) al servidor web Apache. El servidor s'encarregarà de rebre aquestes peticions, realitzar les operacions necessàries que s'explicaran mes endavant, i respondre al client enviant-li la informació correctament estructurada (amb HTML) i adornada (amb CSS).

A continuació es mostra un petit esquema il·lustrant l'explicació anterior:



4.2 HTTP

Com ja s'ha dit, el client del sistema serà un navegador web qualsevol. Això implica que la comunicació amb el servidor es realitzarà mitjançant el protocol HTTP (Hypertext Transfer Protocol). HTTP és un protocol de nivell aplicació (sobre *TCP/IP*) per a sistemes d'informació distribuïts, cooperatius i hipermèdia (la informació pot ser expressada en diferents formats: text, gràfic, àudio, vídeo, etc.). HTTP va ser dissenyat per la W3C i la IETF amb l'objectiu de ser utilitzat en entorns web, pel que es un protocol ràpid i senzill que permet la transferència de múltiples tipus d'informació de forma eficient i ràpida.

D'entre les serves característiques principals destaquem:

- HTTP és un protocol sense estat, és a dir, no guarda cap informació sobre connexions anteriors. Cada petició és tractada per al servidor web com una crida atòmica e independent.
- Permet la transferència d'objectes multimèdia.
- HTTP és unidireccional, la interacció només és possible quan el client crida al servidor. Cal dir però, que actualment es fa ús d'una nova tecnologia anomenada Comet, "germana" de AJAX que serveix per trencar aquesta característica, el servidor pot ser l'iniciador de la comunicació cap al client (p. ex: el chat del Gmail).
- Existeixen tres comandes bàsiques que un client pot utilitzar per comunicar-se amb el servidor:
 - GET: S'utilitza per recollir informació del servidor mitjançant la URL. S'utilitza sempre que es prem un enllaç o s'escriu directament en una URL. Té l'inconvenient que al passar-se informació per la URL és fàcilment visible per l'usuari i si es vol, es pot modificar (que depenent com estigui implementada l'aplicació pot ser un forat de seguretat).
 - POST: A diferència de l'anterior mètode, aquest disposa de més seguretat,

no es visible per part de l'usuari però tampoc és totalment robust front a possibles atacs. Se sol utilitzar aquest mètode per enviar informació al servidor mitjançant formularis.

- HEAD: Sol·licita informació sobre un objecte (fitxer): mida, tipus, data de modificació, etc. Es usat sovint pels gestors de caché de pàgines o per servidors proxy, per conèixer si es necessari o no, actualitzar la còpia que es manté d'un fitxer.

Cada cop que l'usuari tecleja una web en el seu navegador (p. ex: www.quamarket.com), aquest realitza una consulta DNS per esbrinar la direcció IP associada al nom www.quamarket.com, per exemple 217.126.62.245. Aleshores intenta establir una connexió TCP al port 80 (el predeterminat dels servidors web).

Quan aquesta connexió s'ha establert, el navegador envia la petició HTTP sol·licitant la pàgina indicada teclejada per l'usuari. Com que no s'ha escrit una pàgina en concret, el navegador suposa que es vol veure la pàgina principal, que sol anomenar-se `index.php`, `index.html` o `index.asp`.

El servidor retorna la resposta (plana principal) al client que feia la petició. Per acabar, es tanca la connexió.

Aquest procés es repeteix per cada accés al servidor HTTP. Cal dir, que actualment tot aquest procediment s'ha millorat. A diferència d'abans, una connexió es pot mantenir activa durant un cert període de temps per tal de ésser utilitzada en successives transaccions i no carregar amb tantes peticions al servidor. Aquest mecanisme se l'anomena *Keep Alive* i es habitual el seu ús per part de navegadors i servidors. Com ja sabem, avui en dia Internet es troba força saturat i es imprescindible fer servir característiques com aquestes que ajudin a subsanar-lo poc a poc.

Per cada transacció amb el servidor HTTP, aquest retorna un codi informant a l'usuari sobre el resultat de l'operació. Aquests codis poden aparèixer al client quan es produeix un error tant en el client com en el servidor (ex: Error 404).

4.3 Llenguatges de programació

A continuació s'explicaran els llenguatges de programació que s'utilitzaran per desenvolupar el projecte. Donat que PHP el que fa es treure codi HTML, es veu convenient explicar una mica el llenguatge HTML malgrat no hi hagi cap plana estàtica en el sistema.

4.3.1 HTML

Perquè diverses persones es comuniquin és necessari que aquestes parlin un mateix idioma. El llenguatge que utilitzen les computadores que estan connectades a Internet és HTML.

L'HTML, Hyper Text Markup Language és el llenguatge de marques de text utilitzat normalment en la WWW (World Wide Web). Va ser creat al 1986 pel físic nuclear Tim Berners-Lee a partir dues eines existents: El concepte d'Hipertext (conegut també com link o enllaç) permet connectar dos elements entre si i el SGML (Llenguatge Estàndard de Marcació General) serveix per a col·locar etiquetes o marques en un text que indiqui com ha de veure's. HTML no és pròpiament un llenguatge de programació com C++, Visual Basic, etc., sinó un sistema d'etiquetes. HTML no presenta cap compilador, per tant algun error de sintaxi que es presenti aquest no ho detectarà i es visualitzarà de la manera que ho entengui.

Aquest llenguatge es va crear en un principi amb objectiu divulgatiu. No es va pensar que la web arribaria a on es troba avui en dia, de forma que, HTML es va crear sense tenir en compte tots els possibles usos que se li donaria. Es per aquest motiu, que HTML s'ha anat modificant i millorant a través de diferents versions. Aquesta evolució tan anàrquica de HTML ha suposat tota una sèrie d'inconvenients i deficiències que s'han anat subsanant poc a poc mitjançant la inserció d'altres

tecnologies capaces d'organitzar, optimitzar i automatitzar el funcionament de les webs, com CSS o Javascript que més endavant es comentarà.

L'entorn per a treballar HTML és simplement un processador de text (com ara *Vim* o *Kate*). Cal dir però que actualment trobem eines més visuals (inclús online) per a crear planes web de manera molt ràpida e intuïtiva.

La descripció d'un fitxer HTML es basa en especificar una estructura lògica del contingut (títols, paràgrafs, llistes, etc) així com els diferents efectes que es poden atorgar al text (cursiva, negreta, subratllat,etc) i deixar que la presentació final del hipertext es realitzi mitjançant un programa especialitzat, és a dir, un navegador.

A continuació es mostra un petit exemple:

```
<HTML>
<HEAD>
<TITLE>T&iacute;tol de la plana</TITLE>
</HEAD>
<BODY>
<!-- Això es un comentari-->
<H1>Aqu&iacute; un t&iacute;tol</H1>
<P><IMG SRC="logo.jpg" "ALIGN="MIDDLE" ALT="Logo" /></P>
<P> Link cap a la plana principal <A HREF="http://www.quamarket.com">
aqu&iacute;</A>
<HR />
</BODY>
</HTML>
```

Aquest exemple seria interpretat per el navegador com segueix:

Aquí un títol



Link cap a la plana principal [aquí](#)

El fitxer comença per una etiqueta <html> indicant que el programa ha de interpretar el fitxer com un HTML. Com s'observa a l'exemple, per cada etiqueta d'obertura de bloc s'ha de posar també una etiqueta de tancament (a excepció de certes etiquetes). Dins d'un document HTML hi tenim sempre les etiquetes de <head> i <body/>. Dins l'etiqueta <head> es sol ubicar el títol de la plana web, els links cap a les fulles d'estil CSS o arxius de Javascript així com el codi que es vol executar en primera instància (té prioritat al que trobem dins de <body>). En l'etiqueta <body></body> es on hi ha tot el contingut de la plana en qüestió. S'hi pot incloure text, imatges, formularis, taules, llistes, Flash, etc.

4.3.2 PHP

El llenguatge PHP és un llenguatge interpretat de propòsit general. Va ser dissenyat originalment per a la creació de pàgines web dinàmiques. S'usa principalment en interpretació del costat del servidor encara que actualment pot també ésser utilitzat des d'una interfície de línia de comandes o en la creació d'altres tipus de programes.

PHP és un acrònim recursiu que significa PHP Hypertext Pre-processor (o Personal Home Page Tools). Va ser creat originalment per Rasmus Lerdorf al 1994, no obstant això la implementació principal de PHP és produïda ara per "The PHP Group" i serveix com l'estàndard per a PHP al no haver-hi una especificació formal. Publicat sota la PHP License, la Free Programari Foundation considera aquesta llicència com programari lliure.

PHP és un llenguatge interpretat de propòsit general àmpliament usat i que està dissenyat especialment per a desenvolupament web i pot ser inserit dins de codi HTML. Que un llenguatge sigui interpretat (no compilat) significa que està dissenyat per a ser executat per mitjà d'un intèrpret, cosa que no ocorre amb els llenguatges compilats. PHP pot ser desplegat en la majoria de servidors web i en gairebé tots els sistemes operatius i plataformes sense cost algun. PHP es troba instal·lat en més de 20 milions de llocs web i en un milió de servidors. La versió exacta amb la que s'ha treballat aquest projecte ha estat PHP 5.2.6.

Com ja es va comentar a l'inici d'aquesta memòria, el projecte es realitzarà en la seva totalitat amb software lliure. Es per tant que des d'un principi ja es va descartar de treballar amb ASP (de Microsoft), competència directa de PHP. Es també cert, que PHP no es l'única alternativa gratuïta a programació web, citaré a continuació algunes avantatges que em van motivar a treballar amb ell:

- La seva sintaxi és semblant a la de C, que és amb la que he après a programar
- Corba d'aprenentatge curta
- Portabilitat entre plataformes
- Amplia documentació
- Facilitat d'ús

Un altre avantatge és el poder manipular imatges al vol. Donat que certs usuaris de l'aplicació podran pujar imatges al servidor, i no se'ls pot obligar a pujar imatges en diferents mides (imatge en miniatura e imatge ampliada) per a no fer-se lent i pesat, PHP ens ajudarà en aquesta tasca. Sols serà necessari modificar la configuració de l'arxiu *php.ini* activant la llibreria GD (llibreria gràfica) i gràcies al projecte PHP Thumb (aplicació de codi lliure per generar *thumbnails*) s'aconseguirà de tenir les imatges amb la mida que es vulgui, amb la qualitat i proporcions especificades.

Més endavant d'aquesta memòria, en la part d'implementació, es mostraran petites porcions de codi PHP, per això s'ha cregut oportú mostrar un petit exemple de com funciona aquest:

Codi PHP	Resultat de l'execució
<pre data-bbox="194 1093 989 1435"> <?php echo "<p>Això és un bucle:</p>"; for (\$i=0;\$i<10;\$i++){ echo "<p>" . \$i . "</p>"; } echo "<p>Fí del bucle</p>"; ?> </pre>	<pre data-bbox="1091 1025 1305 1496"> Això és un bucle: 0 1 2 3 4 5 6 7 8 9 Fí del bucle </pre>

Com s'observa, el codi d'una aplicació (o porció) PHP sempre va delimitat entre les marques **<?php** i **?>**. Tot allò que no es trobi entre aquestes marques serà ignorat i enviat per la sortida estàndard directament. En l'exemple, s'ha fet ús d'un bucle tal qual es faria en C per exemple o en Java. El codi que finalment expulsa aquesta aplicació es codi HTML. L'avantatge que el codi PHP s'executi dins del servidor és que ningú podrà accedir a ell, només podran veure el codi font com a molt i fer-se una idea aproximada de que fa el programa.

4.3.3 Javascript

Javascript es un llenguatge de programació creat per Netscape a finals de l'any 1995 amb l'objectiu d'integrar-se amb HTML i facilitar la creació de pàgines web. No s'ha de confondre Java amb Javascript ja que Java es un llenguatge complet que permet crear aplicacions independents mentre que Javascript funciona com una extensió del HTML.

A l'igual que amb PHP, Javascript és també un llenguatge interpretat, és a dir, que no requereix compilació, utilitzat principalment en pàgines web, amb una sintaxi semblant a la del llenguatge Java i el llenguatge C. S'utilitza per a millorar la presentació i el comportament de les nostres pàgines web.

El seu codi es pot trobar de tres maneres diferents:

- Scripts immediats: s'executen només carregar la pàgina i es troben dins de les etiquetes `<body></body>`.
- Scripts diferits: es carreguen amb la pàgina però no s'executen fins que l'usuari fa una determinada acció (premer un botó, una tecla, etc). Van dintre del `<body>`.
- Scripts híbrids: es defineixen tant dins del `<head>` com dins del `<body>`.

Per a incloure un script en les nostres pàgines hem d'usar l'etiqueta `<script></script>`.

Una característica important de Javascript a diferència de la majoria de llenguatges es que no disposa de tipus de dades. Per a definir una variable simplement utilitzarem la paraula clau *var*. També es important saber que Javascript s'executa en el costat client, cosa positiva si es busca alleugerir la càrrega de l'aplicació.

A l'igual que Java i PHP, Javascript és un llenguatge orientat a objectes ja que disposa d'herència, encara que aquesta es realitzi seguint el paradigma de programació basada en prototips, ja que les noves classes es generen clonant les classes base i estenent la seva funcionalitat. Actualment, tot navegador modern interpreta codi Javascript integrat dintre de les pàgines web.

Entre les típiques accions que Javascript pot realitzar podríem destacar tres variants:

- Efectes especials sobre la plana web per tal de crear continguts dinàmics com elements en moviment, canvis de colors, canvis d'opacitat, etc.
- Resposta a accions de l'usuari (esdeveniments) com per exemple moure el ratolí, prémer determinada tecla, fer un clic de ratolí, etc.
- Verificar les dades entrades per l'usuari en un formulari abans d'ésser enviat al servidor. En el cas d'aquest projecte, alhora de donar d'alta un nou producte o fer un canvi de nom o qualsevol acció on aparegui un formulari web, es farà ús de Javascript per a verificar la informació. Cal dir, però, que el mòdul Javascript d'un navegador es pot desactivar i per tant convé normalment fer de nou la comprovació dins del servidor per evitar que ens entrin codi no desitjat.

De les característiques més importants de Javascript és que aquest es realitza dins del propi document HTML. Això significa que en la plana es barregen diferents llenguatges de programació. Per tal de que es puguin barrejar diferents llenguatges, s'ha de notificar això mitjançant delimitadors que separin les etiquetes HTML de les instruccions Javascript. Aquests delimitadors són les etiquetes `<script></script>`.

En el desenvolupament d'aquest projecte es farà ús de la llibreria Javascript *jquery* per a la portada. En ella s'hi podrà observar una petita animació realitzada amb una llibreria Javascript anomenada Cycle que precisa de la ja anomenada *jquery* per funcionar. Cycle permet realitzar *slideshow's* de fotografies, és a dir, una petita presentació de fotografies sense tenir que fer ús de Flash (que seria lo habitual anteriorment), un detall purament estètic. També s'utilitza *jquery* per a fer l'autocompletat del cercador de l'aplicació. Finalment dir, que el menú principal de seccions utilitza Javascript per tal d'amagar i mostrar les subseccions de que es compona (encongir-se/expandir-se). Concretament, es treballarà amb la versió 1.8 de Javascript (publicada al juny del 2008).

4.3.4 AJAX

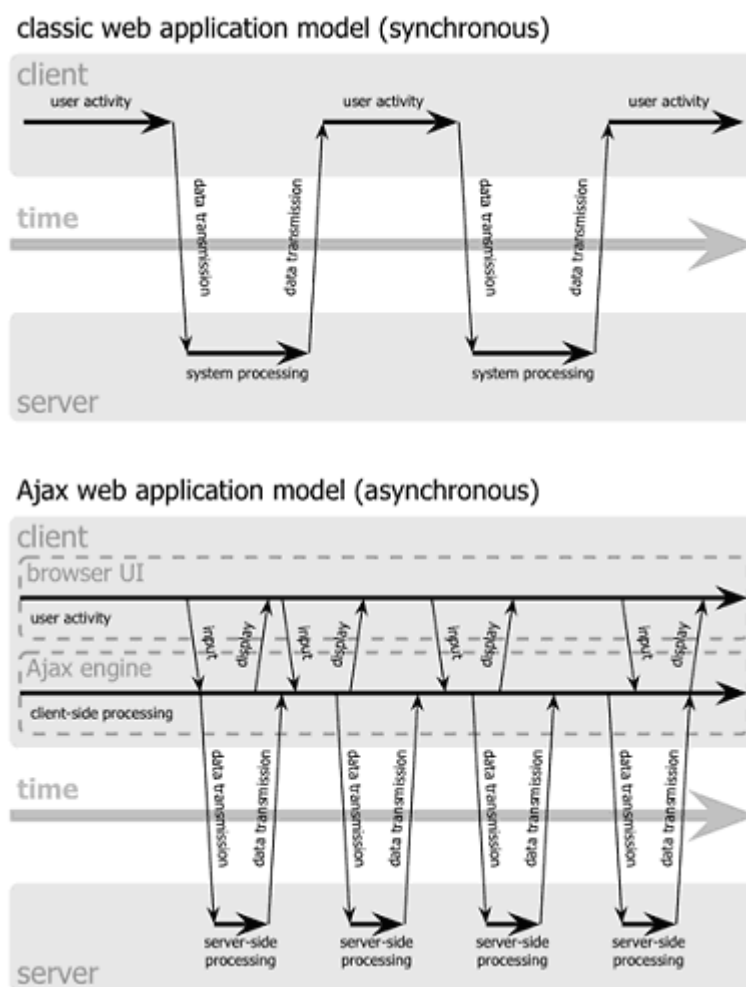
AJAX, significa Asynchronous Javascript + XML. Es tracta d'un conjunt de tecnologies que s'uneixen per a oferir un servei, que incorpora:

- Presentació basada en estàndards usant XHTML i CSS
- Interaccions dinàmiques mitjançant DOM (Document Object Model)
- Intercanvi i manipulació de dades mitjançant XML i XSLT
- Recuperació de dades asíncrona usant XMLHttpRequest
- Javascript com nexa d'unió de tot això

Bàsicament, la principal virtut d'AJAX rau en la potència que se li pot extreure al treball asíncron de peticions al servidor. Estem acostumats a un model d'interacció sincrònica basada en clic – petició - presentació, clic – petició - presentació. Amb AJAX la interacció passa a ser asíncrona. Cada vegada que es fa clic no necessàriament s'estableix una connexió amb el servidor.

AJAX fa d'intermediari entre el servidor i l'usuari, anticipant peticions de dades al servidor, de manera que quan l'usuari fa un clic determinat, AJAX ja té llestes aquestes dades i els mostra directament, sense haver de tornar a fer una petició al servidor i la seva conseqüent espera.

En el següent gràfic s'intenta mostrar la diferència entre el funcionament tradicional en vers al que usa AJAX:



En un desenvolupament tradicional, els clients interaccionen mitjançant l'ús de formularis que fan una petició. Aquesta és rebuda pel servidor, processada i retornada en forma de resposta al client. D'aquesta manera, de vegades s'envia i

reenvia tota una pàgina web esperant tant sols una dada, amb la qual cosa, l'ample de banda es desaprofitat, ja que molta de la informació continguda en la resposta, ja existia originalment en la primera pàgina.

En un desenvolupament basat en AJAX és possible enviar peticions al servidor enviant solament la informació necessària. El servidor rebrà la petició, la processarà i regressarà només les dades necessàries a la pàgina original. Amb tot això, lògicament, es guanyarà bastant en ample de banda i per tant, en temps de resposta del servidor. Al final, el rendiment i funcionament de les aplicacions es veuran millorades. A mode de exemple, aplicacions importants que utilitzen AJAX son Gmail, Facebook, GoogleMaps, etc.

4.3.5 XAJAX

Si hem intentat alguna vegada treballar amb AJAX per programar una pàgina web, haurem comprovat que la tasca es complica bastant, havent de crear diferents trossos de codi en diferents llenguatges de programació i en diferents arxius. Tot això pot provocar mals de cap o pàgines amb codis difícils d'entendre i de mantenir, cosa que aniria en contra del requisit no funcional de mantenibilitat que es va marcar a l'inici d'aquest projecte. Per sort, eines com ara XAJAX ens poden ajudar bastant a facilitar el treballar amb AJAX.

XAJAX és una biblioteca de codi obert per a PHP que permet crear de manera fàcil i simple aplicacions web basades en AJAX utilitzant a més HTML, CSS, i Javascript. Les aplicacions desenvolupades amb Xajax poden comunicar-se asíncronament amb funcions que es troben en el servidor i així actualitzar el contingut d'una pàgina sense tenir que recarregar-la novament, la seva última versió és la 0.5 Final i es amb la que es treballarà el projecte.

Per tal de poder treballar amb XAJAX, solament es precisa crear una instància de l'objecte XAJAX. Aquest objecte s'encarregarà de tot el processament a través de XAJAX. En segon lloc, cal registrar quines funcions seran les que volem que usin AJAX a través de l'objecte XAJAX, això es fa mitjançant el mètode `xajax->register()`. Finalment, totes les respostes seran processades utilitzant el mètode `xajax->processRequest()`.

Per exemple:

```
//es registra l'objecte
$xajax = new xajax();

//es registra una funció que ens interessi que treballi amb AJAX
$xajax->register(XAJAX_FUNCTION, 'finalitzar_comanda');

//li diem que ja ho pot processar
$xajax->processRequest();
```

Cal dir, que en l'actualitat es troben ja diverses classes de biblioteques i *frameworks* de AJAX disponibles per als desenvolupadors d'aplicacions web. Algunes d'elles són Prototype i ScriptAcuclos. El perquè m'he decantat per XAJAX i no per aquestes altres, és que per treballar amb les altres llibreries es precisen uns coneixements sòlids de Javascript, en canvi, XAJAX és una biblioteca purament centrada en PHP. Amb XAJAX, la utilització de AJAX amb PHP es fa molt més senzill, només es necessita escriure sentències curtes i simples en Javascript, el que suposa un avantatge enfront d'altres solucions.

Entre d'altres característiques hi podem trobar que:

- XAJAX és compatible amb Firefox, Mozilla, Internet Explorer, Opera, Chrome i Safari

- XAJAX pot ser usat per a actualitzar estils, classes CSS, botons de selecció, caselles de verificació i botons de ràdio o qualsevol altre atribut d'un element
- Cada funció registrada per a ser accessible a través de XAJAX pot tenir diferents tipus de petició.

4.3.6 CSS

Com ja s'ha comentat, HTML està limitat alhora de donar forma a un document. Això es així perquè es va crear fa anys amb un objectiu merament divulgatiu, no es va pensar que podria arribar a on es avui.

Per tal de solucionar aquests problemes, els dissenyadors han desenvolupat tecnologies annexes a HTML, i CSS es un clar exemple. Per contra, aquestes tecnologies sovint donen problemes en les pàgines alhora de visualitzar-les en diferents plataformes.

Finalment dir, que un altre motiu perquè aparegués CSS va ser que el codi HTML es feia complicat d'entendre i crear, ja que en ell hi trobàvem tant les etiquetes que donaven forma com les estètiques.

CSS ens proporciona tres maneres diferents de poder aplicar les seves regles a una plana web:

1. Fulla d'estil externa: és una fulla d'estil que es troba emmagatzemada en un arxiu diferent a l'arxiu on s'emmagatzema el codi HTML de la pàgina web. Aquesta és la manera de programació més potent que trobem, perquè separa completament les regles estètiques de la pàgina HTML de l'estructura bàsica de la pàgina.
2. Fulla d'estil interna: és una fulla d'estil incrustada dintre d'un document HTML.

(la solem trobar dins de l'element <head></head>). D'aquesta manera s'aconsegueix separar la informació de l'estil, del codi HTML. Es pot optar per copiar la fulla d'estil incrustada d'una pàgina a una altra. En general, l'únic cop que s'usa una fulla d'estil interna, és quan es vol proporcionar alguna característica a una pàgina web en un simple fitxer, per exemple, si s'està enviant quelcom a la pàgina web.

3. Estils en línia: és un mètode per inserir llenguatge d'estil de pàgina, directament, dintre d'una etiqueta HTML. Aquesta manera de procedir no és gaire adequada. Incrustar el codi del format dins del document de la pàgina web, a nivell de codi es converteix en una tasca llarga, tediosa i poc elegant de resoldre el problema de la programació de la pàgina. Aquesta manera de treball es podria usar de manera ocasional si es pretén aplicar un formato amb pressa, al vol. No és la manera correcta d'utilitzar-ho, però funciona.

Un dels únics problemes d'aquesta tecnologia es que no tots els navegadors la suporten, hi si ho fan, no tots la solen interpretar igual. Es per tant, que cal utilitzar aquesta tecnologia amb compte, i abans d'entregar el projecte, fixar-se com es veu en cada un dels navegadors estàndards.

La versió de CSS amb la que es treballarà el projecte serà la CSS 2.

4.3.7 XML

Pel que fa la part de les comparatives de preus en el sistema, es farà ús d'una llibreria gratuïta que proporciona uns gràfics amb Flash força atractius. No es toca per res codi Flash (ActionScript) però per tal de poder-li proporcionar les dades de com es formaran les gràfiques s'usarà XML, es per tant que s'ha cregut oportú explicar una mica que és XML. El sistema, a partir de codi PHP crearà XML a partir d'informació extreta de la base de dades.

XML és una tecnologia envoltada d'altres tecnologies que la complementen i la fan molt més gran i amb unes majors possibilitats. És un llenguatge d'etiquetatge extensible molt simple, però també molt estricte que juga un paper fonamental en l'intercanvi d'una gran varietat de dades. És molt similar a HTML però la seva funció principal és descriure dades i no mostrar-los com és el cas de HTML. XML és un format que permet la lectura de dades a través de diferents aplicacions.

En el món d'Internet hi trobem una gran varietat d'aplicacions i sistemes totalment diferents entre si, que precisen comunicar-se d'alguna manera, és aquí on entre en joc XML.

Un fitxer XML s'escriu d'una manera similar a HTML, a la capçalera s'ha de posar `<?xml version="1.0"?>`, i a la resta de document es deuen escriure etiquetes com les de HTML però amb la diferència que en aquest cas no son etiquetes prefixades, sinó que som nosaltres mateixos qui podem escollir com es diran. Les etiquetes s'escriuen anidades, és a dir, unes dins d'altres, per exemple:

```
<ETIQ1>
  ...
  <ETIQ2>...</ETIQ2>
  ...
</ETIQ1>
```

Qualsevol etiqueta pot disposar de múltiples atributs com per exemple:

```
<ETIQ1 atribut="valor" ..>
```

Qualsevol comentari que inserim en el codi es farà de la mateixa manera que en HTML.

A mode d'exemple, es mostrarà un petit tros de com seria el document XML a crear:

```
<chart_data>
  <row>
    <null/>
    <string>Mitjana general de preus dels súpers</string>
  </row>
  <row>
    <string>Super A</string>
  </row>
  ...
</chart_data>
```

A simple vista, aquest codi pot semblar HTML, el que passa és que mentre que HTML es preocupa per formatejar les dades amb l'objectiu d'estructurar la informació a mostrar, XML es preocupa per estructurar la informació que pretén emmagatzemar.

Resumint, XML serveix per a descriure informació i HTML serveix per a donar-li format i presentar-la a través d'un navegador. O sigui que XML no és ni serà mai un reemplaçament de HTML sinó un complement per gestionar la informació i separar-la del que es el format.

Una dada important és que quan parlem de XML parlem de documents ben formats (well formed). Mentre en HTML podem ser (molt) desordenats:

<P><i>. Aquest text es vorà en negreta i en cursiva</i>
, en XML tenim que respectar les etiquetes. Totes deuran ser anidades correctament:

```
<TagA><TagB>Contingut</TagB></TagA>
```

Un altre cosa molt important és que les etiquetes són "case sensitive" de manera que no és el mateix
 que
, les majúscules importen. És per tant, que XML està ajudant a que siguem més ordenats i respectem més el llenguatge que, degut a l'excessiva tolerància dels navegadors ha anat decaient.

Actualment es fa ja ús de XHTML, una mena de barreja d'idees de XML i HTML, on l'objectiu es crear codi més net, de manera més estricta, per aconseguir arribar en un futur no molt llunyà a les famoses webs semàntiques, webs que seran interpretades i avaluades automàticament per màquines de processament, que potenciarà de forma extraordinària les cerques mitjançant llenguatge natural.

4.3.8 SQL

SQL (Structured Query Language) és el llenguatge estàndard de comunicació amb bases de dades relacionals. Es un llenguatge normalitzat que ens permet treballar en combinació amb qualsevol tipus de base de dades (MySQL, PostgreSQL, Oracle, Informix, etc).

El fet que sigui un llenguatge estàndard no significa que sigui idèntic per a cada base de dades. En efecte, determinades bases de dades implementen funcions específiques que no han de funcionar necessàriament en unes altres (per exemple, en SQL Server no disposem de la funció LIMIT cosa que en MySQL si).

Apart d'aquesta universalitat, SQL posseïx dues característiques molt d'agrair. D'una banda, presenta una potència i versatilitat notables que contrasta, per una altra, amb la seva facilitat d'aprenentatge.

La primera versió de SQL va ser desenvolupada a IBM per Andrew

Richardson, Donald C. Messerly i Raymond F. Boyce a principis dels 70'. IBM va patentar la seva versió de SQL al 1985, mentre que el llenguatge SQL no va ser formalment estandarditzat fins al 1986 per l'American National Standards Institute (ANSI), com SQL-86.

Una base de dades està composta per taules que emmagatzemen registres catalogats en funció de diferents camps o atributs. Donat que una base de dades treballa amb tot tipus d'informació (des de text fins a imatges), és important especificar quin tipus de valors s'introduiran de manera que a posteriori es pugui facilitar la cerca d'aquests. Existeixen tipus de dades comunes, a banda dels diferents tipus que pot introduir cada producte com ara: enters, reals, binaris, blob (imatges), booleans, etc.

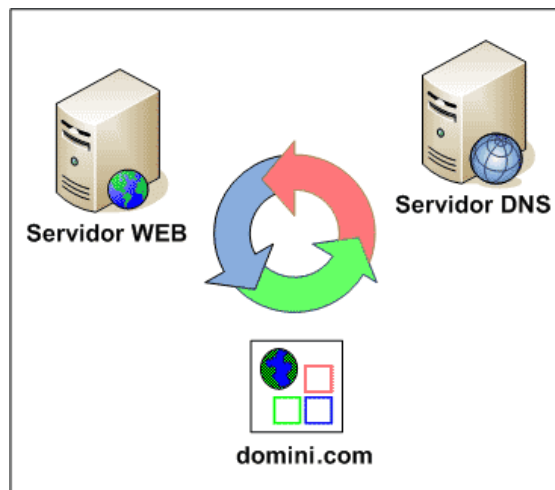
Durant la realització del projecte es farà ús de bases de dades amb taules i registres que s'aniran omplint mitjançant codi PHP des de el panell de control de l'aplicació. Per tal d'optimitzar les consultes en l'aplicació optimitzada , el que s'intentarà serà seguir les següents pautes:

- Disposar d'un índex en cada taula per tal de agilitzar les cerques
- No utilitzar SELECT * ja que el gestor ha de llegir primer l'estructura de la taula abans d'executar la sentència i això gasta recursos
- Utilitzar exclusivament els camps que es precisin
- Evitar en la mesura del possible la clàusula LIKE
- Ajustar al màxim la longitud dels camps per tal de no desaprofitar espai

S'aconsella utilitzar aquestes indicacions si la base de dades es grossa, això ens pot ajudar a reduir bastant el temps de cerca.

4.4 Servidor web

Per tal que la web funcioni precisem els serveis d'un servidor web. A continuació es mostra un petit esquema del seu funcionament:



Els passos que es segueixen per tal de poder veure la plana en el nostre navegador és:

1. Un usuari tecleja a la barra d'adreces del seu navegador la URL `http://www.domini.com` on "http" és el protocol utilitzat per a parlar amb `www.domini.com`, el nom del servidor.
2. El navegador es comunica amb el servidor DNS per tal de traduir el nom `www.domini.com` i convertir-lo en una adreça IP la qual és usada per a connectar al servidor web.
3. El navegador crea una connexió amb el servidor amb aquesta adreça IP, normalment en el port 80.
4. El navegador envia una petició GET al servidor preguntant-li pel fitxer a mostrar.
5. El servidor envia el text HTML de la plana web sol·licitada al navegador.

6. El navegador llegix el codi HTML que se li ha lliurat i el traduïx d'una forma llegible.

Un servidor web no és més que un programa que s'executa de forma contínua en un ordinador (també s'utilitza el terme per a referir-se a l'ordinador que ho executa), mantenint-se a l'espera de peticions per part d'un client (un navegador) i que contesta a aquestes peticions de forma adequada, servint una pàgina web que serà mostrada en el navegador o mostrant el missatge corresponent si es detecta algun error.

Els programes més difosos per a organitzar un servidor web son: Apache, Microsoft IIS, TOMCAT, GLASSFISH i JBOSS.

En el projecte s'utilitzarà un servidor Apache 2.2 per tal de poder executar-hi planes PHP, a continuació s'explicarà breument que és i com funciona.

4.4.1 Apache

El servidor HTTP Apache és un servidor web HTTP de codi obert per a plataformes Unix, Windows, Macintosh i altres. Implementa el protocol HTTP 1.1 i la noció de lloc virtual (poder tenir varis espais web amb un sol servidor). En torn al 1995 quan es va començar a desenvolupar aquest servidor, es va basar bàsicament amb el popular NCSA HTTPd 1.3, però més tard es va reescriure tot per complet. A mode anecdòtic, l'origen del nom Apache, va ser idea del seu creador, Brian Behlendorf, va triar aquest nom perquè volia que tingués la connotació d'alguna cosa ferma i enèrgica però no agressiva, i la tribu Apache va ser l'última en rendir-se al que aviat es convertiria en govern de EEUU, i en aquests moments la preocupació del seu grup era que arribessin les empreses i "civilitzessin" el paisatge que havien

creat els primers enginyers d'Internet.

El servidor Apache es desenvolupa dintre del projecte HTTP Server (httpd) de l'Apache Programari Foundation.

Apache presenta entre altres característiques missatges d'error altament configurables, bases de dades d'autenticació i negociat de contingut, però va ser criticat per la falta d'una interfície gràfica que ajudi en la seva configuració.

Des de 1996, Apache, és el servidor HTTP més usat. Va arribar la seva màxima quota de mercat en 2005 ocupant el 70% dels llocs web en el món, no obstant això, ha sofert un descens en els últims anys.

4.5 Sistema Gestor de Base de Dades

Arribats a aquest punt, s'explicarà el SGBDs que s'utilitzarà durant el projecte. Un SGBD es un tipus de software molt específic dedicat a servir d'interfície entre la base de dades, l'usuari i les aplicacions que l'utilitzen.

Donat que el projecte ha de ser tot fet mitjançant software lliure s'ha decidit treballar amb MySQL, a continuació es descriurà una mica i el perquè l'he escollit.

4.5.1 MySQL

MySQL és un sistema de gestió de base de dades relacionals, multifil i multiusuari amb més de sis milions d'instal·lacions. MySQL AB desenvolupa MySQL com programari lliure en un esquema de llicenciament dual. D'una banda ofereix sota una llicència GNU GPL MySQL per a qualsevol ús compatible amb aquesta llicència, però per a aquelles empreses que vulguin incorporar-lo en productes privatis han de comprar a l'empresa una llicència específica. Està desenvolupat en la seva major part en ANSI C.

Al contrari de projectes com Apache, on el programari és desenvolupat per una comunitat pública i el copyright del codi està sota el poder de l'autor individual, MySQL és propietari i està patrocinat per una empresa privada, que posseïx el copyright de la major part del codi, es per això que poden vendre-ho a empreses privades.

La seva popularitat com aplicació web està molt lligada a PHP, que sovint apareix en combinació amb MySQL. MySQL és una base de dades molt ràpida en lectura quan utilitza el motor no transaccional MyISAM, però pot provocar problemes d'integritat en entorns d'alta concurrència alhora de realitzar modificacions. En aplicacions web hi ha baixa concurrència en la modificació de dades i en canvi l'entorn és intensiu en lectura de dades, el que fa a MySQL ideal per a aquest tipus d'aplicacions.

Cal dir, que la desavantatge d'utilitzar taules MyISAM, és que no es pot gaudir de les claus forànies i les accions en cascada en cas d'eliminació o actualització. Per tant es farà ús de taules InnoDB que es suportat també per MySQL i si que ens ofereix el tenir claus forànies en contra de disminuir una mica la velocitat en les consultes.

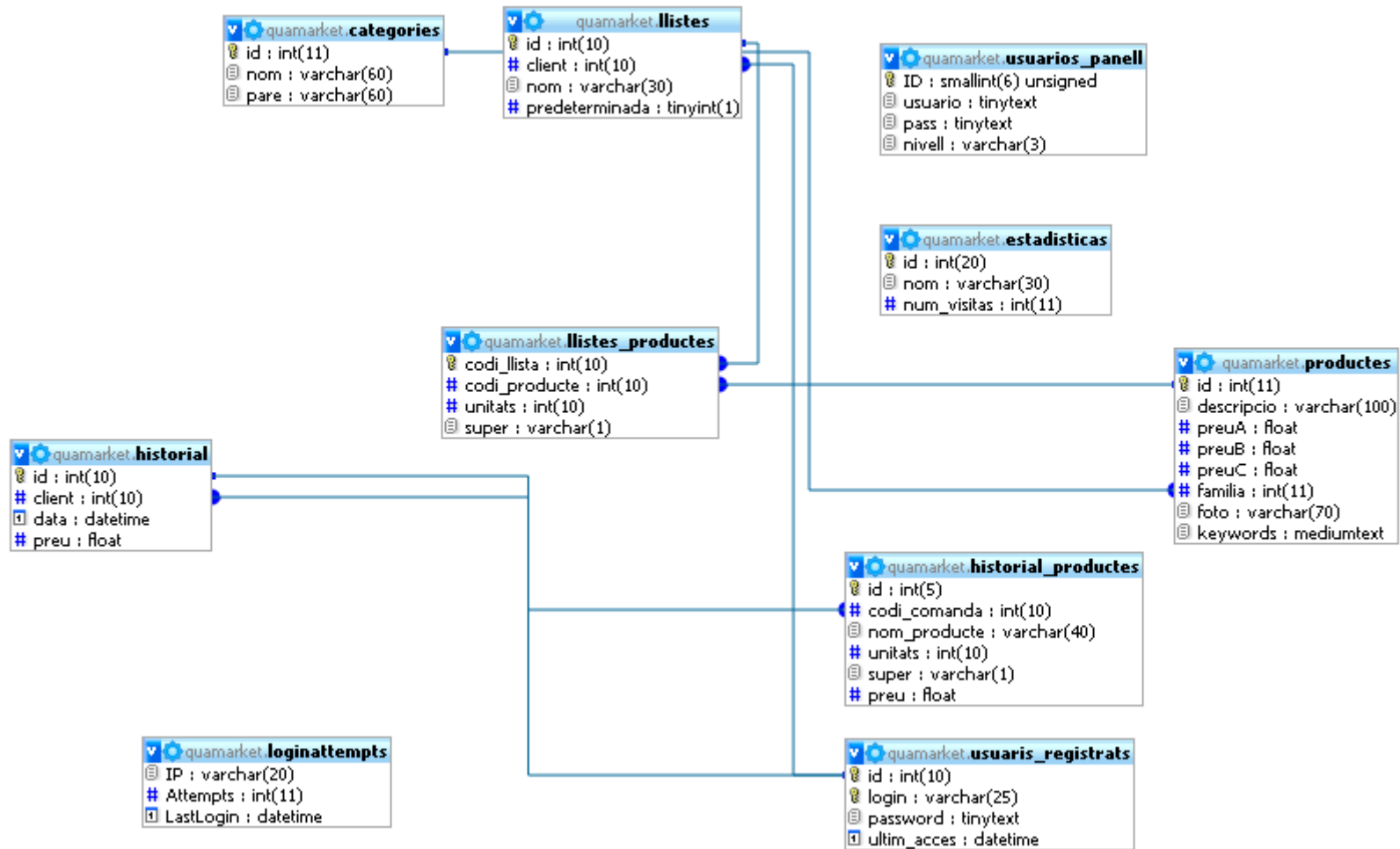
El perquè s'ha escollit MySQL es bàsicament per lo ja descrit, avui en dia MySQL és la base de dades de facto en combinació amb PHP. També es va pensar en postgresql. A diferència de MySQL, postgres és realment bo quan la base de dades amb la que es tracta es molt gran, sent capaç d'ajustar-se al nombre de CPUs i a la quantitat de memòria que posseïx el sistema de forma òptima. Per contra, consumeix bastants més recursos que MySQL. Per aquestes dues raons es va preferir usar MySQL, perquè el projecte en un principi tampoc precisarà una base de dades gran (mes endavant qui sap), i perquè, com es va comentar en els requisits no funcionals de hardware, el sistema hauria de funcionar amb qualsevol servidor

per vell que fos (mínim un Pentium II). També cal dir que Postgres es de 2 a 3 cops més lent que MySQL. Amb tot es pot treure la conclusió que el que no té un, ho té l'altre, s'ha de saber triar per cada projecte quin convé més en funció dels requisits funcionals i no funcionals que s'hagin marcat en un principi.

Per tal de poder treballar amb més facilitat, es farà ús de l'eina phpMyAdmin. Aquesta aplicació funciona via web, esta creada en PHP sota llicència GPL, ens estalviarà de tenir que treballar amb la base de dades via línia de comandes sinó via gràfica que facilitat enormement la feina.

Ja per acabar, i per veure una de les potents característiques que ofereix phpMyAdmin, es mostrarà un petit esquema del què són les relacions (claus forànies) generat automàticament per phpMyadmin un cop fixades totes les relacions entre taules.

PFC PLANA WEB D'AJUDA A LA DECISIÓ EN LA COMPRA DEL SUPERMERCAT



5 Disseny

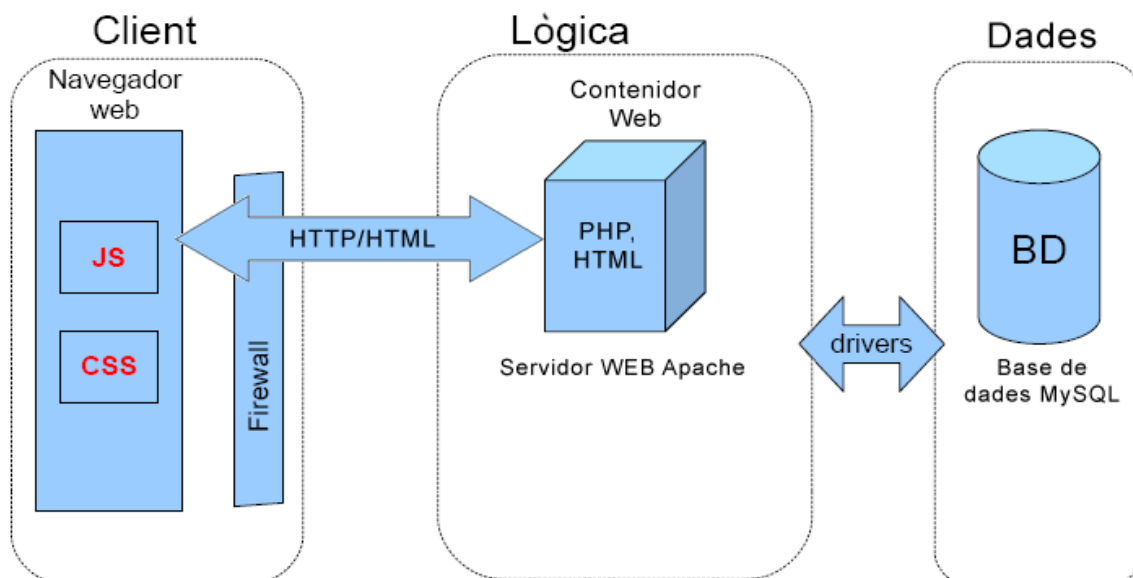
Durant l'etapa de disseny s'explica i documenta com el sistema complirà amb els requeriments descrits durant la fase d'anàlisi.

Concretament, començaré explicant l'arquitectura del sistema i tot seguit quins patrons crec que són els més adients per a desenvolupar el sistema. Més endavant es plasmarà mitjançant un diagrama conceptual les relacions que hi ha entre les diferents classes del sistema i les seves cardinalitats.

Tota aquesta informació detallada serveix perquè un equip de programadors puguin començar la fase de desenvolupament del software.

5.1 Arquitectura del sistema

En el següent esquema es mostra l'arquitectura en que es basa el projecte. A partir d'aquest, es comentarà quin és el flux que realitza una petició al sistema.



Quan un usuari clica per exemple sobre “Afegir un producte al carretó”, el propi navegador de l'usuari controla mitjançant Javascript que es compleixin unes determinades restriccions. Si aquestes es compleixen aleshores s'envia la petició mitjançant un paquet HTTP cap al servidor Apache en el nostre cas. El que fa aquest servidor es gestionar la petició entrar en la capa lògica i emmagatzemar els canvis (si escau) en la capa de dades, en la base de dades. Si tot ha anat correcte aleshores el servidor Apache es l'encarregat de retornar un altre petició cap al client amb informació de que la seva acció ha conclòs amb èxit. Se li ho pot notificar mitjançant Javascript o refrescant la plana (si no es fa servir AJAX) i mostrant la resposta via HTML decorat amb CSS.

5.2 Disseny de l'aplicació

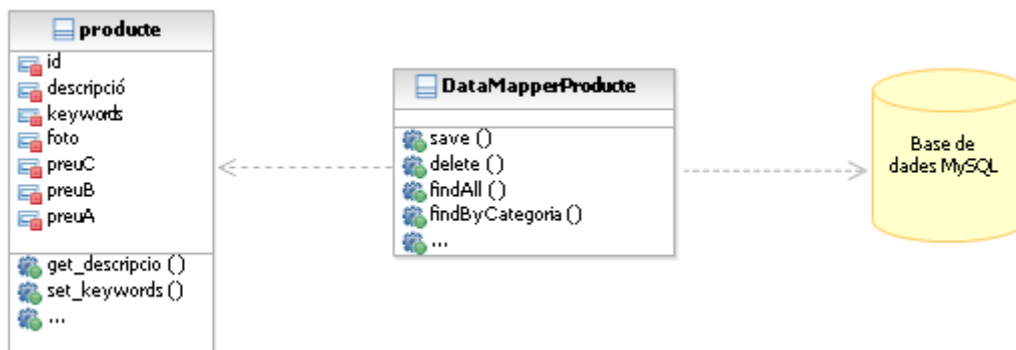
En aquest apartat explicaré quins patrons de disseny s'utilitzaran durant a l'implementació del sistema. Com que el sistema es programarà mitjançant orientació a objectes s'ha decidit que el sistema utilitzarà un patró DataMapper i un Adapter per accedir a la base de dades.

5.2.1 Patró DataMapper

Els objectes i les bases de dades relacionals tenen diferents mecanismes per a estructurar les dades. Moltes parts d'un objecte, com ara col·leccions o herències no hi són presents en les bases de dades relacionals. Si es precisa moure informació entre aquests dos conceptes (objectes i base de dades), aquesta transferència de dades es pot arribar a convertir en quelcom complex i farragós si hi ha moltes dades implicades. Es podria dir que el patró Data Mapper afegeix una capa addicional que el que fa es separar els objectes que es troben en memòria de la base de dades, és a dir, desacobla. La responsabilitat d'un Data Mapper es la de transferir fàcilment informació d'una banda a l'altre (de la capa de dades fins la capa lògica).

Gràcies als Data Mapper, els objectes que es troben en memòria no precisen saber que hi ha una base de dades present, ni tampoc precisen conèixer codi SQL ni com s'estructura la base de dades. Dit d'un altra manera, la nostre capa de negoci no es “casa” amb cap tecnologia de persistència, els objectes de la capa lògica només es preocuparan d'això, de la lògica del sistema i res més.

A mode d'exemple posaré un Data Mapper del sistema a crear:



Com es pot veure en l'exemple, un Producte disposa d'un conjunt d'atributs i d'un conjunt de funcions. Per afegir un nou producte, el que farem per exemple serà cridar des de la capa lògica a la funció “save” del Data Mapper Producte, no ens farà preocupar-nos de res més, d'això ja s'encarregarà el datamapper.

Per cada objecte que tinguem en el sistema, tindrem un Data Mapper associat a ell. Això en un principi aportarà feina addicional però a la llarga si hi ha canvis s'aguirà disposar d'aquest patró.

A mode comparatiu d'aquest patró de disseny, en Java hi tenim el popular sistema “Hibernate”.

5.2.2 Patró Adapter

El patró Adapter ens serveix per a convertir un objecte d'un tipus en un objecte d'un altre tipus. Típicament, els programadors utilitzen aquest procés per embolcallar diferents funcions en una de sola. El patró Adapter es una manera fàcil de netejar i de reutilitzar codi en altres llocs.

En el fons, el que fa el patró es quelcom semblant al patró anteriorment explicat, crear una interfície diferent per un objecte donat, si per exemple hi ha canvis en la interfície del llenguatge usat (externs a nosaltres), sols modificant l'Adapter, tot el codi seguirà funcionant d'igual manera. Dins una funció del Adapter hi poden haver tantes funcions del llenguatge com es desitgi, la idea és “adaptar-nos” una funció existent, ampliar-la o modificar-la al nostre gust.

Malgrat que en aquest projecte aquest patró no es fonamental, crec que es interessant aplicar-lo ja que si en un futur es pretén modificar el sistema i es vol canviar de bases de dades per exemple (de MySQL a PostgreSQL per exemple), només canviant un arxiu ja estaria resolt. En el projecte, es crearà un sol Adapter i serà per accedir a la base de dades. Tindrà dues operacions, una per executar “queries” (o consultes) amb retorn de resultats (consultes a la base de dades) i un altre per “queries” sense retorn (inserir, modificar, eliminar). Per exemple, la funció per executar queries amb retorn seria:

```
function executeWithResult ($sql) {  
  
    include "panell/constants.inc.php";  
  
    $conectar = mysql_connect ($host, $user, $passwd) or die (mysql_error ());  
  
    mysql_select_db ($db, $conectar) or die (mysql_error ());  
  
    $id = mysql_query ($sql);  
  
    if ($id==0) {  
        //debug  
        echo "Problemas con la query_with: $sql";  
        exit;  
    }  
  
    while ($fila=mysql_fetch_assoc ($id)) {  
        $res[]=$fila;  
    }  
  
    return $res;  
}
```

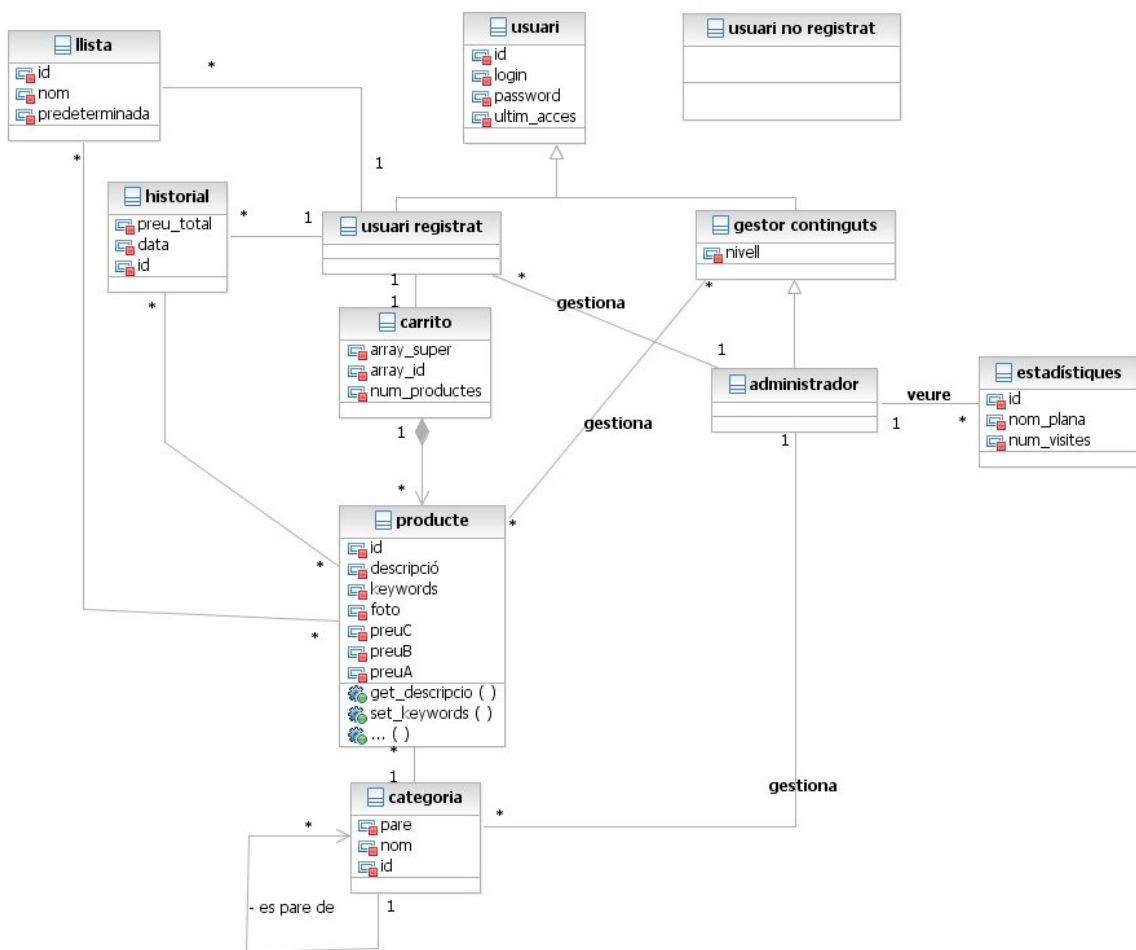
El que es fa en aquest codi es embolcallar un conjunt d'operacions que s'haurien d'anar posant en diferents parts del codi del sistema en una sola funció (executeWithResult). Per a usar aquesta funcionalitat des de qualsevol part del codi només faria falta cridar a `Objecte::executeWithResult("...")` i ens retornaria tots els resultats d'una consulta a MySQL, en aquest cas.

És important recordar que, ja que es programa de forma modular el sistema, cal aprofitar els diferents patrons de que disposem per petits que siguin, perquè a la llarga, ens poden treure molta feina de sobre ja sigui per canvis externs a nosaltres (canvis en la interfície del llenguatge) o en afegiment de noves funcionalitats.

5.3 Diagrama conceptual

Per poder veure les classes que compren el sistema i com interaccionen entre sí, he trobat adient crear un model conceptual de les classes que he utilitzat.

Un diagrama conceptual ens serveix per a veure quines relacions mantenen les diferents classes de que es disposa així com la cardinalitat que pot haver-hi entre elles.



5.4 Disseny de la web

Abans d'explicar com s'implementarà la plana web, es comentarà el disseny que es seguirà en fer-la. La plana principal, estarà disposada per una taula dividida en marcs com es mostra a continuació:

Menú principal de l'aplicació		
Menú amb submenús de les seccions de productes	Contingut de la secció escollida	Desglòs de la cistella
Peu de pàgina		

La totalitat del disseny mostrat es dinàmic a excepció del menú principal i el peu de pàgina. Això significa que un cop entregada l'aplicació, l'usuari pot anar modificant el contingut d'aquestes finestres com desitgi, mitjançant un panell de control. Aquesta propietat es molt important a dia d'avui i ho aconseguim mitjançant, en aquest cas, PHP. Les planes estàtiques estan ja passant a un segon pla, ja que un cop realitzades, si no hi ha modificacions futures per un entès en la matèria, aquella plana queda intacte per sempre. És important recordar, que actualment els cercadors, ja saben emmagatzemar planes dinàmiques (a excepció de Flash).

El menú lateral esquerra que es veu en la figura superior mostra les diferents seccions (com ara Begudes) i les seves subseccions (com ara Refrescants, Infusions,etc).

Aquest menú té una altura màxima de 3, és a dir, pot haver-hi una categoria pare de la que hi pengen fills i d'aquests i pengen "nets". Cal dir que generar aquest menú, és potser la part més costosa a nivell computacional del que hi ha en la plana principal, més tard s'explicarà el perquè en la part d'implementació.

Pel que fa al marc principal de la figura (el central), allí és on es mostra, un cop escollida una secció del lateral esquerre la informació dels productes associats a aquella secció. Això s'aconsegueix passant-li al marc central un paràmetre indicant l'índex de la secció. Per tal d'aconseguir una aplicació més eficient, es farà us d'AJAX en aquesta part, amb això el que s'aconsegueix és no haver de generar de nou tota la plana cada cop que es selecciona una secció del menú, ja que tocaria sinó refrescar tota la pàgina i com ja s'ha dit, crear el menú lateral es costós (a nivell computacional).

Pel que fa al marc de la dreta, allí es mostrarà en tot moment la llista feta fins aleshores. Cada cop que s'afegeixi un nou producte del marc central apareixerà a l'instant al marc dret. També es realitzarà mitjançant AJAX per la mateixa raó. Si es decideix eliminar un producte de la llista allí també es podrà fer.

Moltes de les accions que es podran fer en tots els marcs estaran controlades per Javascript, amb això ens assegurem que no ens entrin dades que no interressi (p. ex.: que no ens entrin un valor no numèric com a unitats en la llista). Es clar que des de qualsevol navegador es pot deshabilitar Javascript, però donat que la majoria de la plana funciona sota AJAX, no funcionaria casi res.

6 Implementació

Un cop arribats a aquest punt, significa que ja es té força clar el que farà i el que no farà el sistema (anàlisis i especificació), i com ho farà (la part de disseny). En aquest capítol s'explicaran les consideracions, a nivell d'implementació, que s'han realitzat durant el desenvolupament del projecte. Es descriuran al complet els aspectes que es consideren més rellevants del sistema. Si es desitja veure la implementació al detall es recomana veure els fitxers fonts que hi ha en el CD que acompanya aquesta memòria.

Login al sistema

Quan un usuari es connecta al sistema, automàticament es comprova que aquest existeixi, si es així aleshores es comprova que la contrasenya proporcionada sigui la seva. Cal dir que al estar la contrasenya encriptada en la base de dades mitjançant l'algorisme MD5, la contrasenya rebuda de l'usuari s'encripta abans de comparar-la amb l'emmagatzemada. Si tot això es correcte, se li assigna aleshores una variable de sessió *autenticat* per a comprovar posteriorment la seva autenticitat, un altre variable de sessió amb el seu nom d'usuari i un altre amb el seu codi d'usuari. Aquest codi d'usuari ens servirà més endavant per a veure les seves llistes de la compra, els seus historials, etc.

Donat que HTTP es un protocol sense estat, és a dir, que cada petició es troba en el seu estat inicial o dit d'un altre manera, no te memòria entre peticions, s'ha decidit utilitzar les variables de sessió per emmagatzemar informació relativa a l'usuari que es connecta, aquestes variables venen a ser una alternativa a les famoses "cookies" però molt més segures. Una cookie es un fitxer de text pla que amb qualsevol editor de text es pot modificar.

En el codi que es mostra a continuació es pot veure la implementació del login al sistema, al principi s'inicia la sessió amb *session_start()* (imprescindible si s'usen

variables de sessió). Després es comprova si ja existeix la variable de sessió “autenticat”, si aquesta ja existeix no es fa res més. En cas contrari, es comprova el que s'ha dit al principi d'aquest apartat.

Cal dir que a l'inici de cada un dels fitxers de codi, es passa aquest control, per evitar que algú pugui entrar al sistema si es sap una ruta directe a un dels fitxers.

També es pot veure com es guarda constància de l'hora de connexió de l'usuari. Si tot va bé se'l redirecciona cap a main.php, sinó es torna cap enrere i se li indicarà amb una alerta que el login a fallat.

```
<?php
session_start();
require_once "datamappers/DataMapperUsuaris_registrats.php";
require_once "datamappers/DataMapperllistes.php";

if(!isset($_SESSION['autenticat']))(

    //això sols dona un resultat perquè login és un camp UNIQUE
    $resultat = DataMapperUsuaris_registrats::findByLogin($_POST['login']);

    if(count($resultat)==1){
        if(md5($_POST['password']) == $resultat[0]->get_password()){
            $_SESSION['autenticat'] = "1";
            $_SESSION['usuari'] = $resultat[0]->get_login();
            $_SESSION['id_usuari'] = $resultat[0]->get_id();

            $llistapred = DataMapperllistes::findLlistaPredeterminada($resultat[0]->get_id());
            $_SESSION['llista_actual'] = $llistapred;

            //guardem el moment de la connexió
            $resultat[0]->set_ultim_acces(date("Y-m-d H:i:s"));
            $ok = DataMapperUsuaris_registrats::save($resultat[0]);

            header("Location: main.php");
        }
        else{
            header("Location: index.php?err=0");
        }
    }
    else{
        header("Location: index.php?err=0");
    }
}
?>
```

El codi per fer login al panell de control es molt semblant però es comprova

també un codi (captcha) per evitar que bots malignes intentin accedir al panell i corrompre el sistema.

Menú de les seccions

El menú on es mostren les seccions de que disposa el sistema, és un menú creat mitjançant PHP, JavaScript, CSS i MySQL. És un menú costós de generar a nivell computacional. L'estructura que s'ha seguit en la base de dades per a mantenir una jerarquia adient es la que segueix:

Campo	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Extra
id	int(11)			No		auto_increment
nom	varchar(150)	latin1_swedish_ci		No		
pare	varchar(150)	latin1_swedish_ci		No		

Cada secció pot tenir o no un pare associat. Si no en té cap el camp pare es deixa a -1, altrament en el camp pare s'hi posa el índex del pare. El menú s'ha creat a partir de llistes (element de HTML). Per tal de crear el menú, el que s'ha realitzat ha estat un algorisme recursiu e iteratiu ahora:

```

function prepara_menu($pare){

    $categoria = DataMapperCategories::findByPare($pare);

    if(count($categoria)>0){

        if($pare==-1) $sol.= "\n\n<ul id='listaMenu'>\n";
        else          $sol.= "\n\n<ul>\n";

        foreach($categoria as $categ)
        {
            $fills          = DataMapperCategories::findPareCount($categ->get_id());
            $num_articles   = DataMapperProductes::findByFamiliaCount($categ->get_id());

            // si jano te mes fills li posem el link
            if($fills==0)
                $sol.= "\t<li>
                    <a href='#' onClick='xajax_omple_main(\"".$categ->get_id()."\",0);return false;'>
                        $categ->get_nom()."
                    ($num_articles)</a>";
            else
                $sol.= "\t<li>".$categ->get_nom();

            $sol.= prepara_menu($categ->get_id());

            $sol.= "</li>\n";
        }
        $sol.= "</ul>\n";
    }
    return $sol;
}

```

Com es pot veure en aquest codi, la funció rep un paràmetre indicant un codi pare. És per tant que la primera crida que es faci serà a *prepara_menu(-1)* perquè ens interessa que els pares predominin primer. Per cada pare cercat es busquen les categories que pengen d'ell i es fa una iteració sobre elles. Podria ser que un dels seus fills fos alhora pare de més categories, si això passa es repetiria el procés (recordem que aquest arbre té una alçada de 3 nivells). Altrament es posa un enllaç que al preme'l, l'usuari vorà els productes de la secció en el marc central.

Per cada categoria es torna a cridar a *prepara_menu()* passant-li el codi de la categoria en qüestió, i així de manera recursiva s'aconsegueix crear l'arbre sencer. A l'inici del codi hi ha un IF() que ens serveix perquè si s'arriba a una secció que no té fills, l'algorisme acaba per aquella branca.

Com informació addicional, el mateix menú indica per a cada categoria filla quants productes hi pengen d'ella, això s'observa on diu (*\$num_articles*).

Fins aquí s'ha explicat el que aporta PHP i MySQL alhora de crear el menú. Pel que fa a Javascript, el que ens ajuda alhora de crear el menú és que es mantinguin ocultes les categories filles fins que l'usuari no cliqui en una categoria pare. Això s'aconsegueix donant un codi únic a cada llista. Tanmateix, al clicar sobre una categoria pare, les altres categories que estiguin expandides, es col·lapsaran per tal de no carregar massa el menú.

Finalment, el que aporta CSS a tot això és la definició dels colors dels enllaços, els marges que hi ha entre ells així com la altura de cada ítem de la llista. Bàsicament, tota la part estètica del menú.

A mode d'exemple he vist oportú posar un tros de codi resultat de tot això:

```
1 <ul id='listaMenu'>
2
3 <li>Aperitius
4 <ul <!--subllista Aperitius-->
5 <li>
6 <a href='#' onClick='xajax_omple_main("67",0);return false;'>
7 Fruits secs (2)</a>
8 </li>
9
10 <li>
11 <a href='#' onClick='xajax_omple_main("68",0);return false;'>
12 Snacks (0)</a>
13 </li>
14 </ul> <!--tanca subllista Aperitius-->
15 </li> <!--tanca llista Aperitius-->
16
17 <li>Basar
18 <ul>
19 <li>...
20
21 ...
22 </ul> <!--tanca llista principal listaMenu-->
```

Un cop PHP ha generat tot el codi del menú, Javascript entra en acció i s'encarrega d'amagar tots els possibles submenús generats, en l'exemple, amagaria el submenú dels Aperitius de la línia 4-14.

Si l'usuari clica sobre la secció d'Aperitius, Javascript s'encarregaria de mostrar-nos la llista (Fruits secs i Snacks), això ho fa mitjançant una propietat de CSS que es diu display (posant-la a 'none' o a 'block' segons convingui).

Llistat dels productes

Quan un usuari selecciona del menú esquerre de l'aplicació una secció se li està passant al servidor Apache una petició on conté un paràmetre1 amb el codi de la secció i un paràmetre2 indicant des de on es crida a la funció:

```
function omple_main($apartat,$type)
```

Això es així perquè a aquesta funció se la crida des de diferents llocs:

```
//$type=0 cerca per seccio estandard  
//$type=1 cerca per text  
//$type=2 cerca producte concret
```

El type 0 es la crida “estàndard”, la que quan es clica una secció s'omple en el marc central els productes d'aquesta. Els altres type's, es criden des de la secció cercador, depenent de si es selecciona un producte en concret ($$type==2$) o si es selecciona una característica concreta ($$type==1$).

Per cada producte es mostra una foto redimensionada mitjançant la llibreria PhpThumb, la descripció d'aquest, els 3 preus dels súpers, un selector d'unitats que es poden entrar al carretó i finalment un “checkbox” que ens indica si el producte pertany a la llista actualment activa de l'usuari. Per defecte es selecciona el super que ofereix el producte més barat d'entre els 3 possibles mitjançant la funció matemàtica “min” que ofereix PHP. Pot ésser que no es tingui el preu d'un super, si es així, el sistema ignora aleshores aquell preu.

Al agregar un producte al carretó, pot ésser que l'usuari per comoditat o per altres circumstàncies prefereixi seleccionar un altre super que el seleccionat per defecte, això ho pot fer mitjançant els “radio buttons” que hi ha sota el preu de cada súper. També es té en compte si l'usuari intenta entrar una quantitat no vàlida com ara les menors de 0 o valors no numèrics.

Si alhora de llistar una secció o producte(s) n'hi ha un que ja es troba en el carretó en aquell moment, es mostraran les unitats actualment demanades i el súper ja escollit.

Carretó virtual

Per tal de mantenir un carretó actiu durant tota la sessió sense haver de fer us de la base de dades, s'ha decidit d'utilitzar un objecte “carretó” dins una variable de sessió. L'objecte en qüestió disposa dels següents atributs:

```
//atributs de la classe
private $num_productes; //nombre de productes totals
private $array_id_prod; //si esta a -1 esta eliminat
private $array_quantitat;
private $array_productes; //conjunt d'objectes
private $array_supers; //a quin super pretenen comprar el producte, per defecte al mes barat
```

Cada cop que s'insereix un producte s'omple cada un dels atributs a la posició *\$num_productes* de cada vector. Contràriament, si es vol eliminar un producte del carretó només caldrà posar la variable *\$array_id_prod* corresponent a -1. Al llistar els productes o calcular el total (en euros) es descartaran els productes amb aquest valor a -1. Dins la classe carretó també hi podem trobar funcions consultores i funcions modificadores de cadascun d'aquests atributs així com una funció per calcular l'import total de la llista.

A l'hora d'afegir o eliminar productes s'utilitza de nou la tecnologia AJAX per tal de disminuir el treball que ha de realitzar el servidor.

Motor de cerca

En el cercador de productes de l'aplicació, ens trobem amb un formulari on entrar-hi el producte o característica a cercar. Cada cop que l'usuari pitja una tecla en el camp de text, s'envia una petició al servidor indicant la paraula escrita fins aleshores. Si la paraula te més d'un caràcter es mostrarà en un desplegable els resultats a l'usuari (si n'hi ha), fins un màxim de 7.

A l'hora de cercar en la base de dades es tenen en compte dos cassos, si la paraula a cercar esta formada només per un tros o per més d'un. Si està formada només per una paraula es fa una cerca com la següent:

```
$query = BDDadapter::executeWithResult("SELECT * FROM productes WHERE  
descripcio LIKE '%$queryString%' OR keywords LIKE '%$queryString%'");
```

La paraula a cercar es troba en la variable *\$queryString*, aquesta es cercarà en la descripció dels productes o en les “keywords” d'aquests (que s'entren alhora de donar d'alta un producte).

En canvi, si la paraula està formada per més d'una part, per cada tros es fa una cerca com la mostrada i s'emmagatzema en un vector. Si una de les parts no es troba en la cerca automàticament ja se li notificarà a l'usuari que no hi ha cap resultat associat. De tots els resultats obtinguts es busca quins productes encaixen amb les diferents parts de la cerca entrada (ja sigui de la descripció o de les “keywords” del producte), és per això que hi hauran resultats que es filtraran per tal de mostrar una cerca més acurada a l'usuari.

Creació dels PDF's

Per a la creació dels PDF's s'ha fet ús de la llibreria FPDF, una llibreria molt coneguda per tot programador PHP. Aquesta llibreria té un conjunt de plugins creats per l'àmplia comunitat d'usuaris que l'utilitza. En aquest projecte s'ha utilitzat un d'aquests plugins per tal de poder generar, a partir de codi HTML un PDF, i no haver-lo de crear des de zero, cosa que implicaria controlar diverses característiques com ara els marges laterals, l'alçada de cada línia, els salts de pàgina, etc. Al ésser una llibreria tant completa té l'inconvenient que és força complexa. El plugin en qüestió s'anomena HTMLParser.

Un cop es mostra la llista final a l'usuari se li dona l'opció d'imprimir la plana utilitzant la característica de qualsevol navegador per al fet i l'opció de guardar-la en PDF. Si l'usuari vol guardar la llista en PDF, el que passa internament, és que se li envia una petició al servidor amb el codi de la llista a generar. Aquest crea una instància (\$p) del HTMLParser, i el que fa es generar el HTML corresponent a mostrar (en l'exemple, això ho trobem en la variable *\$html*), configurar la font, passar-li el HTML a la instància \$p, i realitzar un Output final. A l'usuari se li obrirà, mitjançant el seu visor d'arxius PDF, l'arxiu resultant amb el qual podrà guardar una instància d'aquest si li interessa.

```
$p = new PDFTable();
// set margins
$p->AddFont('vni_times');
$p->AddFont('vni_times', 'B');
$p->AddFont('vni_times', 'I');
$p->AddFont('vni_times', 'BI');

$p->SetMargins(20,20,20);
$p->AddPage();
$p->defaultFontFamily = 'times';
$p->defaultFontStyle = '';
$p->defaultFontSize = 12;

$p->SetFont($p->defaultFontFamily, $p->defaultFontStyle, $p->defaultFontSize);

$p->htmltable($html);
$p->output('llistes.pdf', 'I');
```

7 Planificació i costos

7.1 Diagrama de Gantt

Donada que l'estimació es d'unes 20 hores per crèdit, i que el projecte en té 37,5, s'hauria de fer una mitja de 750 hores. A continuació es mostrarà mitjançant un diagrama de Gantt com es repartiran aquestes hores. El diagrama s'ha realitzat mitjançant OpenProj, una alternativa lliure a Microsoft Project.

	📌	Nombre	Duración	Inicio	Terminado
1		Estudi previ	29 days	15/01/09 9:00	24/02/09 13:00
2		Definició d' objectius	1 day	15/01/09 9:00	15/01/09 13:00
3		Planificació	1 day	16/01/09 9:00	16/01/09 13:00
4		Avaluació tecnològica i elecció	7 days	19/01/09 9:00	27/01/09 13:00
5		Formació en les tecnologies implicades	20 days	28/01/09 9:00	24/02/09 13:00
6		Preparació de l' entorn de treball	2 days	15/01/09 9:00	16/01/09 13:00
7		Anàlisis de requisits	10 days	25/02/09 9:00	10/03/09 13:00
8		Requisits funcionals	4 days	25/02/09 9:00	2/03/09 13:00
9		Requisits no funcionals	6 days	3/03/09 9:00	10/03/09 13:00
10		Especificació	14 days	11/03/09 9:00	30/03/09 13:00
11		Descripció dels cassos d' ús	14 days	11/03/09 9:00	30/03/09 13:00
12		Disseny	24 days	31/03/09 9:00	1/05/09 13:00
13		Definir l' arquitectura de l' aplicació	3 days	31/03/09 9:00	2/04/09 13:00
14		Avaluació de patrons de disseny i elecció	11 days	3/04/09 9:00	17/04/09 13:00
15		Diagrama conceptual	3 days	20/04/09 9:00	22/04/09 13:00
16		Disseny de la plana web	7 days	23/04/09 9:00	1/05/09 13:00
17		Implementació	61 days	4/05/09 9:00	27/07/09 13:00
18		Implementació de la plana	61 days	4/05/09 9:00	27/07/09 13:00
19		Testing	9 days	28/07/09 9:00	7/08/09 13:00
20		Proves unitàries	7 days	28/07/09 9:00	5/08/09 13:00
21		Solventar possibles errors	2 days	6/08/09 9:00	7/08/09 13:00
22		Documentació	53 days	25/02/09 9:00	8/05/09 13:00
23		Memòria	48 days	25/02/09 9:00	1/05/09 13:00
24		Revisió de la memòria	5 days	4/05/09 9:00	8/05/09 13:00
25		Preparació de la presentació	12 days	10/08/09 9:00	25/08/09 12:23

Diagrama de Gantt - part1

PFC PLANA WEB D'AJUDA A LA DECISIÓ EN LA COMPRA DEL SUPERMERCAT

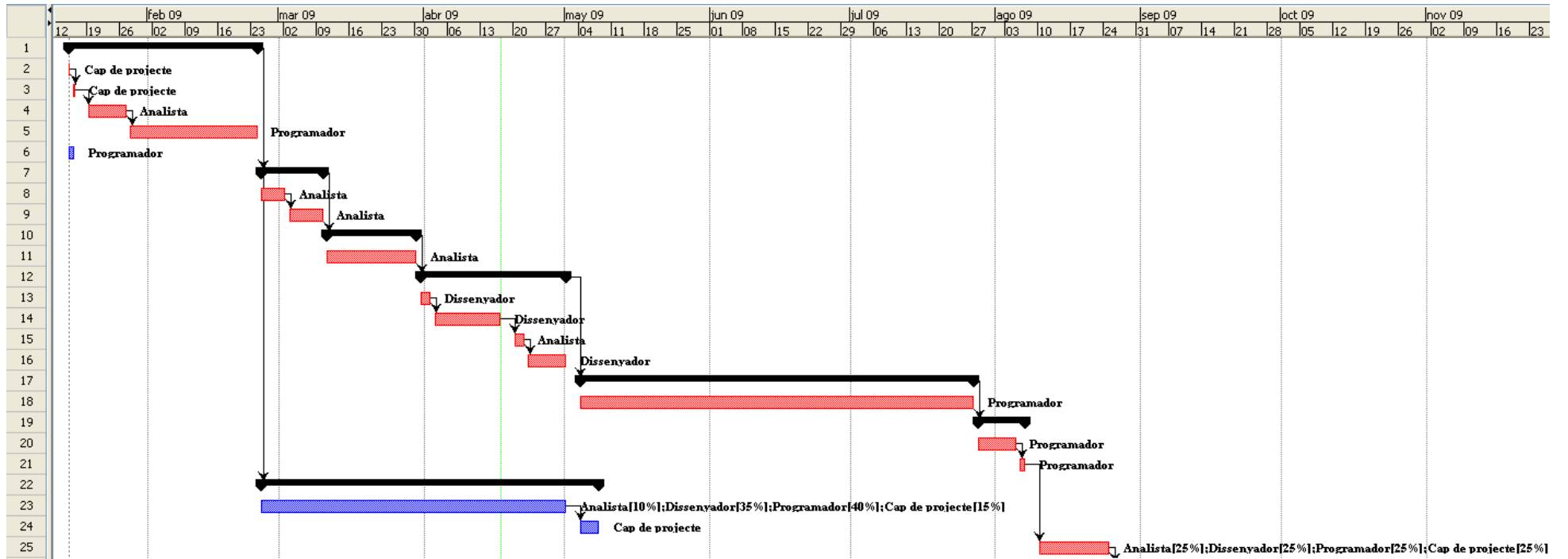


Diagrama de Gantt - part2

Com s'observa en el diagrama, a la mateixa hora que s'ha anat desenvolupant el sistema s'ha fet la documentació. És per això, que hi ha tasques que s'han allargat, ja que, al haver-n'hi que usen els mateixos recursos (perquè es fan en paral·lel), els esforços dels perfils es divideixen per dos.

Les tasques (no documentals) on més temps si dedica son les de formació en les tecnologies en les que es treballarà i la part d'implementació.

7.2 Cost del projecte

Els costos d'un projecte es divideixen en costos de software, costos de hardware i personal. En el cas d'aquest projecte, els costos de software no es tindran en compte ja que es tracta d'un projecte *open source* en la seva totalitat. En quant als recursos hardware tampoc es tindran en compte ja que el hardware necessari s'ha pogut fer tot des de la universalitat. És per això, que només es tindrà en compte els recursos de personal.

Per tal de realitzar el càlcul d'aquests costos s'imputaran les diferents hores a les diferents categories laborals indicades a continuació. Els sous aquí mostrats s'han extret de l'assignatura PGPSI (Planificació i Gestió de Projectes i Sistemes Informàtics). Les hores totals s'han extret del diagrama de Gantt un cop aplicat els recursos pertinents (analista,dissenyador,programador,cap de projecte) a cada tasca.

	Hores	Preu/hora	Preu Total
Cap de projecte	37,85 hores	65,00 €	2.460,25 €
Analista	160,70 hores	40,00 €	6.428,00 €
Dissenyador	111,70 hores	40,00 €	4.468,00 €
Programador	452	30,00 €	13.560,00 €
Total en recursos humans	762,25 hores		26.916,25 €

A continuació s'explicarà les funcions que realitza cada perfil en el projecte:

Cap de projecte: es l'encarregat de coordinar el projecte i els seus integrants. Ha de mirar que el projecte es finalitzi en el marge de temps acordat, sense passar-se del pressupost i buscant un bon clima dins del equip de treball i amb el client. Se li ha atribuït la definició dels objectius, la planificació i un 10% de les hores de documentació.

Analista: és aquell individu que exerceix les tasques d'anàlisi dels sistemes informàtics, amb la finalitat d'automatitzar-los. S'encarrega de realitzar l'avaluació tecnològica, l'anàlisi de requeriments i l'especificació del sistema. Se li han atribuït apart, un 10% de les hores imputades a la documentació.

Dissenyador: el dissenyador s'encarrega de la part de disseny, aquella que comporta l'arquitectura del sistema, els patrons de disseny i el disseny de la plana web. Se li ha imputat també un 20% de les hores dedicades a la documentació.

Programador: és l'encarregat de programar l'aplicació un cop especificada, realitzar les proves (testing), les hores dedicades a estudiar les tecnologies a utilitzar així com un 40% de la documentació de l'aplicació. En la documentació hi trobem el manual d'usuari, també fet pel programador.

A mode comparatiu, es posarà un petit estudi del que hauria suposat haver de fer el projecte amb software privatiu:

Producte	Alternativa open source	Preu del producte
Microsoft Windows XP	Linux Ubuntu	130,00 €
Microsoft Office	OpenOffice	400,00 €
Microsoft Project	OpenProj	500,00 €
Adobe Photoshop CS4	Gimp	985,00 €
Adobe Dreamweaver CS4	Amaya	555,00 €
Total programari estalviat		2.570,00 €

Com s'observa, el preu estalviat és considerable. Es clar que es podria haver afegit bases de dades més potents com ara Oracle però no s'ha vist oportú de posar en la comparativa. També es podria eliminar el Photoshop i el Dreamweaver però aleshores realitzar la web es faria més lent i farragós i s'augmentarien les hores de programació i disseny. Crec que el que es mostra en la comparativa és el software “estàndard” (o habitual) alhora de realitzar una plana web.

8 Conclusions i possibles extensions futures

L'objectiu inicial d'aquest projecte era crear una aplicació web que contemplava tres súpers aprop de casa. El que s'ha intentat fer durant la realització del projecte ha estat preparar l'aplicació per a que en un futur proper sigui fàcilment escalable i es puguin afegir més supermercats i per tant, diferents zones de procedència de l'usuari, ja que, ara per ara, l'aplicació sols té ús en un poble petit (com ara per exemple el meu, Cardedeu) o una comunitat de veïns.

Es pot dir que l'objectiu inicial ha estat superat satisfactòriament. S'ha afegit a més a més funcionalitats típiques de planes d'aquest caire (com ara capraboacasa.com, eroski, mercadona, etc) com son les llistes de la compra (estalvien força temps), el mostrar productes relacionats o la gestió d'usuaris privats.

En conclusió, puc afirmar que la realització d'un projecte informàtic des de zero ha estat força positiva. Durant tota la carrera sempre he cursat assignatures que tractaven temes en concret (amb molta profunditat), aquí en el projecte, en canvi, he pogut plasmar tots aquests coneixements però tots alhora. Per exemple, pel que fa a l'enginyeria del software, vaig cursar ES1 que s'encarregava d'explicar la part d'anàlisis, i ES2 que s'encarregava de la part de disseny. Pel que fa l'implementació, i més concretament, l'orientació a objectes, ho vaig aprendre a PROP, ampliant-ho en PRED. Pel que fa a la planificació i avaluació dels costos ho vaig aprendre en PGPSI, etc. Es per això, que m'agradaria destacar la importància dels coneixements adquirits durant aquests 6 anys de carrera. Gràcies a ells, de manera inconscient sempre busco les solucions més fàcils i eficients possibles a tot problema que se'm presenti, pensant sempre en el disseny global, de manera que faciliti possibles ampliacions del sistema.

Malgrat que alguns dels llenguatges utilitzats ja m'eren coneguts, n'he après d'altres que crec em podran servir en un futur proper, sobretot AJAX i XML.

Agraïments

Ja per finalitzar, m'agradaria donar les gràcies a tothom qui ha fet possible que arribi fins al final d'aquesta llarga carrera, sobretot a la meva família i a la meva parella que m'han donat sempre suport incondicional en moments en que ho veia tot cap per vall. M'agradaria també agrair el suport teòric que m'ha aportat (durant el PFC) en Pere Botella en quant a la part d'enginyeria del software i als ànims del meu tutor i director de projecte, Lluís Pérez Vidal, a tots, gràcies.

Possibles extensions futures

En aquest apartat analitzarem les possibilitats que s'haurien de tenir en compte en el moment de continuar ampliant-lo i millorant-lo:

- Que el nombre de súpers de que disposa l'aplicació sigui fàcilment ampliable (més de tres).
- Portar-lo a Internet, que deixi de ser una aplicació local. Es per tant, que al donar-se d'alta un nou usuari, se li preguntarà per exemple el codi postal d'on es. El sistema calcularà si aquest codi postal disposa de supermercats aprop d'ell (dels entrats al sistema).
- Posar publicitat a la web per tal de poder generar els suficients ingressos com per mantenir almenys el hosting i el domini.
- Crear un sistema de backups de la base de dades des del panell de control del sistema. Aquest crearà un arxiu comprimible i li preguntarà a l'usuari on desitja guardar-lo. Es important no guardar-lo en el mateix ordinador on hi hagi el sistema, ja que si li ocorre quelcom a aquest ens quedem sense sistema i sense aplicació.

- Si el sistema es troba a Internet, és segur que serà visitat per persones no catalanoparlants, per tant, una possible extensió futura seria la de crear l'aplicació multi-idioma.

9 Bibliografia

Patrons de disseny

<http://www.ibm.com/developerworks/opensource/library/os-php-designpatterns/>

DataMapper

<http://martinfowler.com/eaCatalog/dataMapper.html>

“UML y patrones de Craig Larman”

PRENTICE HALL, 1999

“Diseño de páginas Web con XHTML, JavaScript y CSS”

Editorial Ra-Ma

Juan Carlos Orós

“Creación de un portal con PHP y MySQL (2ª edición)”

Editorial Ra-Ma

Jacobo Pavón Puertas

“Ingeniería del software: Diseño II”

Edicions UPC 2003

Carles Farré, Antoni Olivé, Carme Quer

JavaScript

<http://es.wikipedia.org/wiki/Javascript>

<http://www.webestilo.com/javascript/>

XML

<http://www.dat.etsit.upm.es/~abarbero/curso/xml/xmltutorial.html>

AJAX

<http://es.wikipedia.org/wiki/AJAX>

<http://www.desarrolloweb.com/articulos/xajax-libreria-php.html>

Generació dels gràfics en Flash

http://www.maani.us/xml_charts/index.php

Generació d'arxius PDF

<http://www.fpdf.org/>

Generació dels thumbnails

<http://phpthumb.sourceforge.net/>

Informació en general

<http://es.wikipedia.org>

10 Annex

10.1 Glossari

En aquest apartat es definiran els diferents acrònims que apareixen al llarg de la memòria:

Backup: Les còpies de seguretat són un procés utilitzat per a salvar informació, és a dir, un usuari vol guardar una part o tota la informació de la qual disposa en el ordinador fins a aquest moment, realitzarà una còpia de seguretat de tal manera, que ho emmagatzemarà en algun mitjà d'emmagatzematge tecnològicament disponible fins al moment com per exemple cinta, DVD, BluRay, en discos virtuals que proporciona Internet o simplement en altre disc dur, per si posteriorment perd l'informació, poder restaurar-la tota.

Base de dades (BD). Conjunt de dades que pertanyen al mateix content emmagatzemats sistemàticament per al seu posterior us.

Bots: Un bot (diminutiu de *robot*) és un programa informàtic que realitza funcions molt diverses, intentant imitar el comportament d'un humà.

Completely Automated Public Turing test to tell Computers and Humans Apart (CAPTCHA). Es tracta d'una prova utilitzada en computació per determinar quan l'usuari és o no un humà. La típica prova consisteix en que l'usuari introdueixi un conjunt de caràcters que es mostren en una imatge distorsionada que apareix en pantalla. Se suposa que una màquina no és capaç de comprendre e inserir aquesta seqüència de manera correcte per el que solament un humà podria fer-ho.

Cookie: és un fragment d'informació que s'emmagatzema en el disc dur del visitant d'una pàgina web a través del seu navegador, a petició del servidor de la plana. Aquesta informació pot després ésser recuperada per el servidor en posteriors visites.

Cross browsing: es refereix a la habilitat d'una plana web a suportar tots els navegadors. Cross-browsing significa que funciona amb totes les versions de tots els navegadors existents.

Flash: es tracta d'un llenguatge comprat per Adobe al 2005 (fins aleshores pertanyent a Macromedia) per a crear animacions multimèdia. Els seus arxius (també anomenat pel·lícules) tenen format .swf i poden veure's en planes web o mitjançant un reproductor de Flash.

Framework: és una estructura de suport definida, mitjançant la qual un altre projecte software pot ésser organitzat i desenvolupat. Típicament, sol incloure suport de programes, biblioteques i un llenguatge interpretat entre altra software per a ajudar a desenvolupar i unir els diferents components d'un projecte.

Internet Protocol (IP). És un nombre que identifica de manera lògica i jeràrquica a una interfície d'un dispositiu dins d'una xarxa que utilitzi el protocol IP.

Login: és el moment d'autenticació al ingressar a un servei o sistema. En el moment que es fa login en una aplicació, l'usuari entra en una sessió, emprant usualment un nom d'usuari i contrasenya.

Plugin: Es una aplicació que es relaciona amb un altre per aportar-li una nova funcionalitat. Actualment s'utilitza com una forma d'expandir programes de manera modular, de forma que es puguin afegir sense afectar al programa existent ni complicar el desenvolupament del programa principal.

Portable Document Format (PDF). Es un format d'emmagatzemament de documents desenvolupat per Adobe especialment ideat per documents susceptibles a ésser impresos ja que manté el format en qualsevol sistema operatiu en el que es visualitzi.

Query. Consulta de dades que es realitza a una base de dades. És una forma genèrica d'anomenar al que son les insercions, actualitzacions, cerques o eliminació de dades en una base de dades.

Sistema Gestor de Bases de Dades (SGBD). Els sistemes de gestió de base de dades son un tipus de software molt específic, dedicats a servir una interfície entre la base de dades, l'usuari i les aplicacions que l'utilitzen. El seu propòsit general es el de gestionar de manera clara, senzilla i ordenada un conjunt de dades que posteriorment es convertiran en informació rellevant.

SQL injection: es tracta d'una vulnerabilitat informàtica a nivell de validació de les entrades a la base de dades. L'origen és el filtrat incorrecte de les variables utilitzades en les parts del programa amb codi SQL.

TCP/IP: TCP/IP és la base d'Internet, serveix per enllaçar ordenadors que utilitzen diferents sistemes operatius, incloent PC, miniordenadors i computadores centrals sobre xarxes d'àrea local (LAN) i àrea extensa (WAN). TCP/IP va ser desenvolupat i demostrat per primera vegada l'any 1972 pel departament de defensa

dels Estats Units, executant-lo en ARPANET, una xarxa d'àrea extensa d'aquest departament.

Thumbnail: són versions reduïdes d'imatges. S'aconsegueix una millor qualitat d'imatge en comptes de reduir la imatge directament per la propietat "width" de l'etiqueta <img..>.

Unified Modelling Language (UML). (Llenguatge Unificat de modelat). És el llenguatge de modelat de sistemes software més conegut en l'actualitat. És un llenguatge gràfic per a visualitzar, especificar, construir i documentar un sistema. Està respaldat en gran mesura per la OMG (Object Management Group).

eXtensible Markup Language (XML). Es un metallenguatge extensible d'etiquetes desenvolupat per la W3C (World Wide Web Consortium). Permet definir la gramàtica de llenguatges específics. Es per això, que XML no és realment un llenguatge en particular, sinó una manera de definir llenguatges per diferents necessitats.

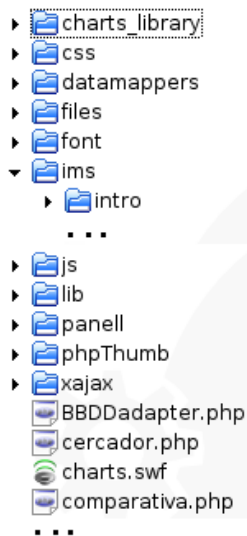
10.2 Manual d'usuari

Instruccions per instal·lar un sistema LAMP + PhpMyAdmin

El primer pas per poder executar l'aplicació és preparar el servidor. La instal·lació es realitzarà sota una distribució Linux Ubuntu (en Gnome). Cal recordar, que si es prefereix es pot contractar un servidor extern que inclogui un servidor Apache amb suport per PHP i MySQL. Altrament, si es vol treballar en un servidor local caldrà seguir els següents passos:

1. Obrir el gestor de paquets synaptic de que disposa Ubuntu (Gnome)
2. En el menú editar clicarem l'opció "Marcar paquetes por tarea".
3. Seleccionarem Lamp Server.
4. Per a instal·lar PhpMyAdmin per tal de facilitar treballar amb la base de dades (encara que no es obligatori perquè funcioni tot) caldrà cercar en el mateix synaptic amb l'opció "Buscar" el paquet phpmyadmin i seleccionar-lo.
5. Clicar en aplicar i tancar synaptic.

En aquest moment i si tot ha anat bé ja es podrà accedir mitjançant qualsevol navegador a la URL <http://localhost>. S'haurà de copiar la carpeta anomenada "quamarket" del CD adjunt a aquesta memòria al directori /var/www/. De manera que l'estructura de directoris dins la carpeta "/var/www/quamarket/" sigui la següent:



Un cop fet aquest pas, caldrà donar permisos a la carpeta “personalitzables” que es troba dins del directori panell. Per a facilitar aquesta acció, es realitzarà en mode gràfic, només caldrà seleccionar la carpeta “panell/personalitzables” amb el botó dret del ratolí i clicar propietats. Allà dins on diu “Permisos” es donarà permís de veure i modificar (tant de owner, group com others) i es clicarà també l'opció de aplicar a totes les carpetes i continguts. Si no es permetés modificar els permisos caldria, des de una consola canviar el propietari de la carpeta amb la següent comanda:

```
sudo chown -R usuari:usuari /var/www/quamarket/
```

Un cop fet això caldrà instal·lar la base de dades del sistema. Mitjançant phpmyAdmin, (entrant a <http://localhost/phpmyadmin>) crearem una base de dades. Si no s'ha posat cap contrasenya, per defecte hi accedirem sols amb l'usuari root.

Per tal de crear la base de dades, escriurem on diu “Crear nueva base de datos” el nom de la base de dades (l'anomenarem quamarket). Un cop fet això, caldrà cercar dins del CD adjunt un arxiu anomenat *quamarket.sql*. Si cliquem ara l'icona que diu SQL a dalt a l'esquerre en phpMyAdmin trobarem l'opció “Importar

archivos”, caldrà indicar-li on es troba l'arxiu `.sql` i a continuació pitjar continuar.

Arribats a aquest punt ja es disposa del servidor LAMP en execució i de la base de dades preparada. Si s'ha configurat MySQL amb un nom d'usuari i contrasenya farà falta fer un pas addicional i configurar els següents 2 arxius amb el nom d'usuari i contrasenya corresponents (per defecte no n'hi ha):

- `aut_config.inc.php` de la carpeta panell
- `constants.inc.php` de la carpeta panell

Arribats a aquest punt ja es té tot el sistema funcionant, només caldrà accedir a la URL:

<http://localhost/quamarket>

Recordatori: Com ja s'ha dit durant la memòria, hi ha ocasions on es precisarà fer ús de les llibreries gràfiques que ens ofereix PHP, concretament GD. És per això, que si no es visualitzen les imatges en miniatura de cada producte caldrà instal·lar la llibreria:

Des de una consola es pot usar la comanda (i reiniciar després): `sudo apt-get install php5-gd`

Pantalla inicial:



En la pantalla inicial hi trobem una presentació d'imatges (estètica) realitzada amb JavaScript. Si se'n volen afegir o treure'n només farà falta afegir fotografies en format JPG, GIF, PNG o BMP a la carpeta `ims/intro/` amb unes dimensions aproximades de 191x143 píxels.

En aquesta pantalla l'usuari podrà donar-se d'alta o accedir-hi si ja es troba registrat en el sistema.

En el primer cas, la pantalla que es vorà serà la següent:



Registrar nou client

NOM D' USUARI

CONTRASENYA

REPETIR CONTRASENYA

NIVELL DE SEURETAT:
0%
Massa curta

MITJA GASTO MANUAL
 €

L'usuari haurà d'omplir el formulari. Si algun dels paràmetres no compleix alguna de les restriccions citades en anteriors apartats no se'l deixarà registrar i se li ho notificarà. Com s'observa a la imatge, el nivell de seguretat anirà canviant de color en funció de la complexitat de la contrasenya introduïda (a cada premuda de tecla s'actualitzarà).

Altrament, si l'usuari ja es troba donat d'alta en el sistema sols haurà d'inserir el seu nom d'usuari i contrasenya i clicar en el botó "Entrar".

Pantalla principal:

Aquest es l'aspecte de la pantalla principal un cop s'ha accedit al sistema. Com s'observa, a dalt a la dreta s'informa del nom d'usuari que ha accedit així com de la data actual. Abaix de tot de la imatge s'aprecia el temps que s'ha trigat per generar tota la plana.

A la dreta de la imatge, si l'usuari hagués afegit a la llista algun producte del menú desplegable de l'esquerre, allí hi apareixeria junt amb les unitats i el súper. Tanmateix es dona l'opció de buidar el carretó i finalitzar la llista si es desitja. Si es finalitza la llista i no hi ha cap producte en el carretó s'avisarà a l'usuari de que no pot continuar. Tanmateix, si es finalitza la llista, el carretó automàticament s'esborrarà.

La pantalla de fi de llista es la següent:

[Imprimir] [Guardar en PDF]

Llistes de la compra

Producte	Unitats	Preu	Preu total
Super A			
Pepsi-Cola	1	3.6 €	3.60 €
Super B			
Nuez grano Borges caixa 60gr	5	1.7 €	8.50 €
Beguda refrescant de Cola	2	1.48 €	2.96 €
Super C			
Nuez Amazonia, 200 gr	4	2.5 €	10.00 €

Terminado

En aquesta pantalla el que es pot fer es imprimir la llista per la impressora o guardar-la en PDF. Si l'usuari sense voler tanca la finestra, més tard la pot recuperar si va a l'historial i troba la llista en qüestió.

Pantalla “Les meves llistes”:



En aquesta pantalla es gestionaran les diferents llistes personals de cada usuari. D'esquerre a dreta les opcions disponibles son:

- Predeterminar una llista. Això significa que a partir d'ara tot producte que s'insereixi a la llista ho farà a aquesta.
- Canviar el nom a una llista.
- Carregar llista al carretó. Si la llista no conté cap element no es procedirà a carregar-la.
- Eliminar una llista i tots els seus elements associats.

Si es vol donar d'alta una nova llista, això es farà mitjançant el boto rodó verd inferior. Al clicar-lo apareixerà un nou camp on introduir el nom de la nova llista. Si no se n'introdueix cap, es notificarà a l'usuari i no es procedirà a la seva alta.

Pantalla "Historial":

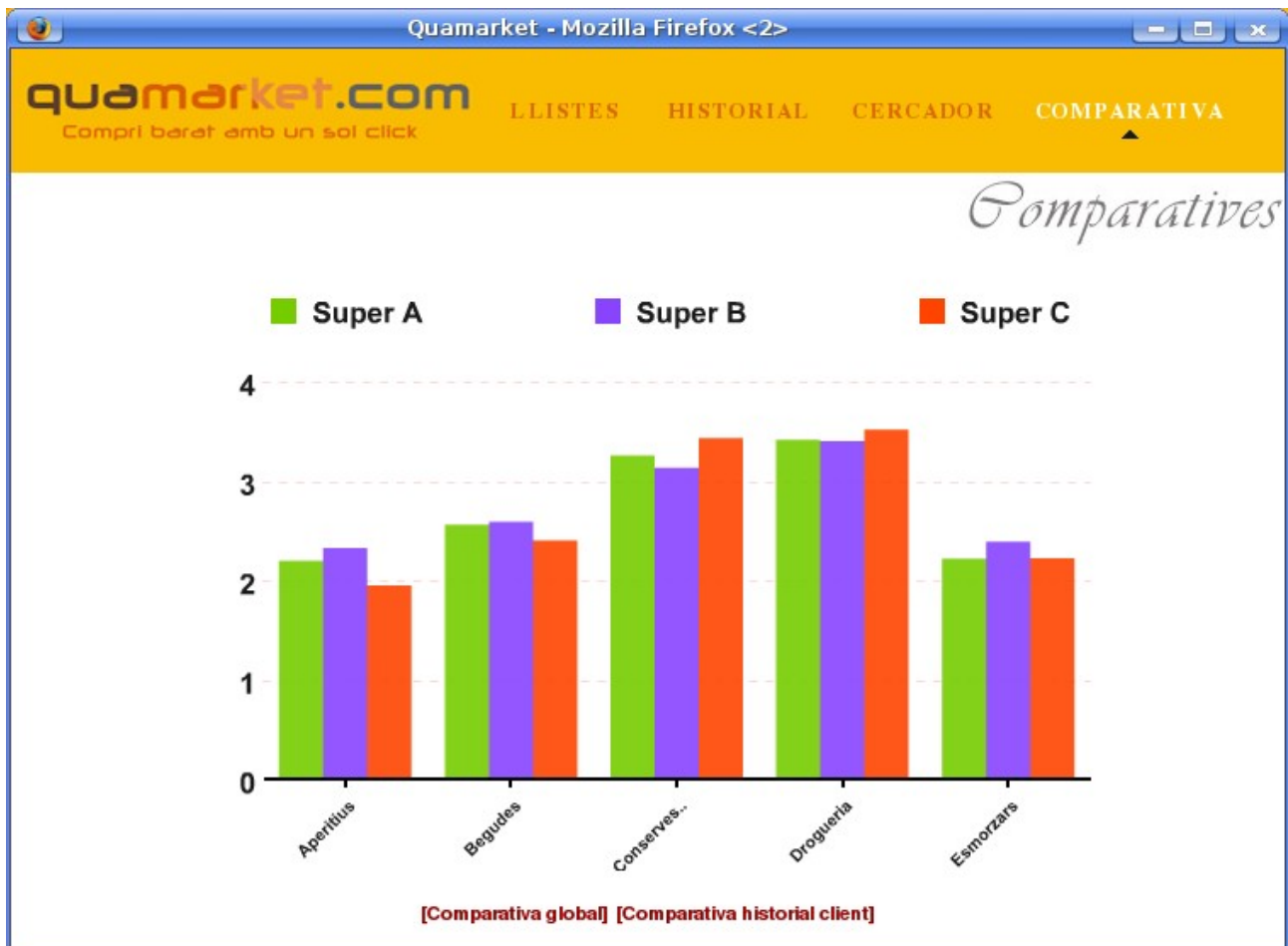
The screenshot shows the 'Historial de compres' page on the Quamarket website. The page has a yellow header with the Quamarket logo and navigation links: LLISTES, HISTORIAL, CERCADOR, and COMPARATIVA. The main content area is titled 'Historial de compres' in a cursive font. Below the title is a table with the following data:

Dia de la compra	Nombre d' articles	Preu total
13-03-2009 a les 18:16	3	15.10 €
23-03-2009 a les 22:57	4	70.00 €
31-03-2009 a les 13:36	2	16.30 €
<ul style="list-style-type: none"> ▪ Café molido descafeinado mezcla BONKA, paquete 250 g [2 unitat(s) al súper C] ▪ Café molido descafeinado BAQUÉ, paquete 250 g [1 unitat(s) al súper B] 		
02-04-2009 a les 22:57	5	14.60 €
08-04-2009 a les 16:13	4	13.10 €
08-04-2009 a les 20:21	6	100.28 €

On the left side of the page, there is a graphic of a hand with a list of items written on the palm, including 'Café molido descafeinado mezcla BONKA' and 'Café molido descafeinado BAQUÉ'.

En aquesta plana l'usuari podrà visualitzar el seu "historial de compres", és a dir, cada cop que es finalitza una llista es guarda una instància en la base de dades. En aquesta pantalla es poden visualitzar totes les instàncies ordenades per data, així com imprimir-les per la impressora estàndard o en PDF (generat al vol). Alhora s'indica la quantitat d'articles de que es composava la llista així com del preu total i l'hora en que es va realitzar.

Pantalla "Comparatives":



En aquesta pantalla l'usuari podrà veure a mode informatiu una comparativa de preus. Concretament tindrà tres opcions:

- Comparativa dels preus en global
- Comparativa dels preus mitjans de cada secció
- Comparativa del seu historial econòmic

Per cada gràfica podrà veure el valor exacte de cada barra si hi passa el ratolí per sobre. Aquests gràfics s'aniran modificant automàticament a mesura que vagi realitzant noves llistes o es vagin afegint productes al sistema.

Pantalla "Cercador":



En aquesta pantalla, l'usuari podrà cercar productes dins la base de dades. Fins que no s'hagi premut més d'una tecla, no apareixeran els possibles productes coincidents amb la cerca. Si es vol es pot prémer sobre un producte en concret o clicar intro quan es tingui una característica escrita.

Panell de control

Si en canvi el que es desitja es gestionar els productes del sistema, gestionar les categories, gestionar usuaris, veure estadístiques del sistema o afegir/treure a nous gestors de continguts, això es deurà fer mitjançant el panell de control, per accedir a ell es necessari d'obrir la següent URL:

http://ip_del_servidor/quamarket/panell/

La plana que es mostrarà tot seguit serà la següent:

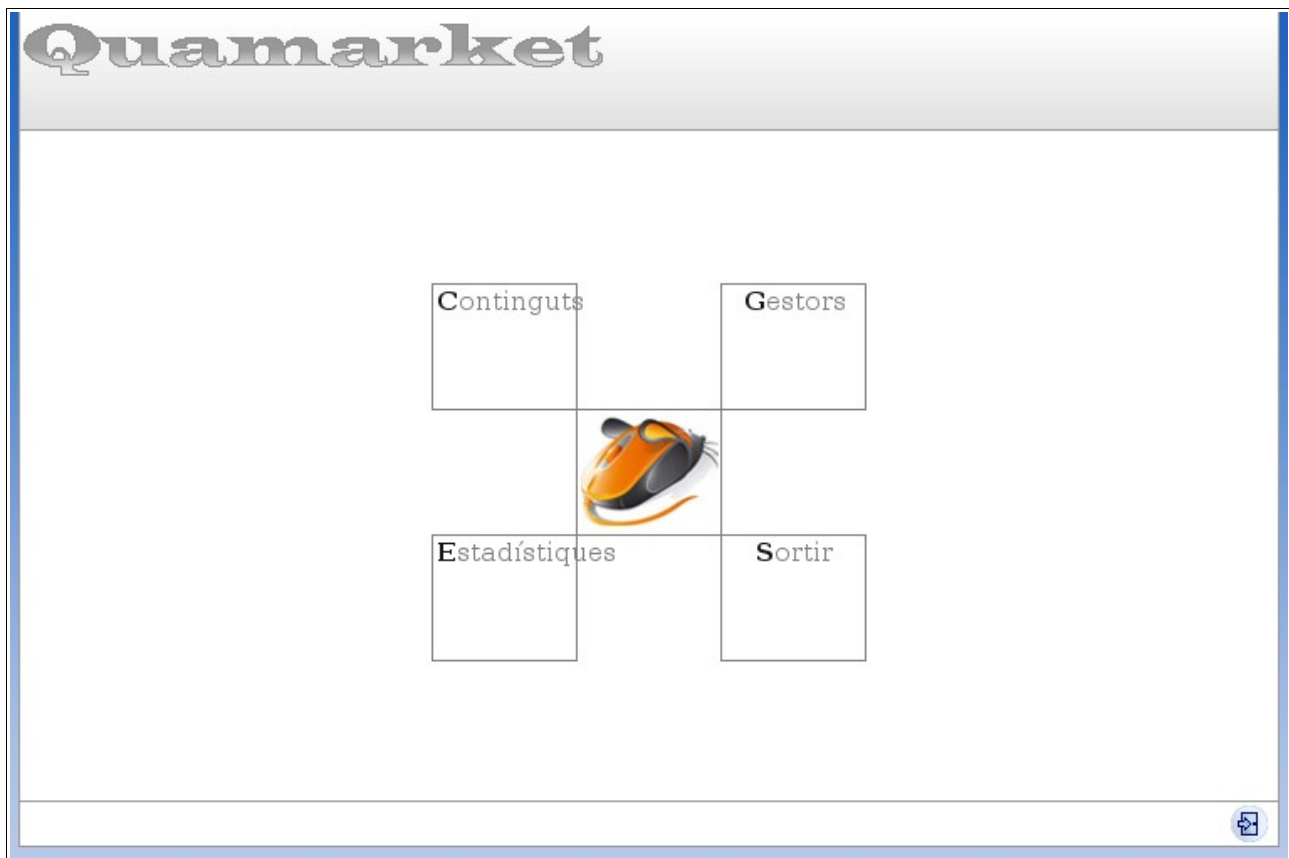
ATENCIÓ: El sistema diferencia entre majúscules i minúscules, procuri no tenir les majúscules actives

Aquí ens trobarem un altre cop amb un menú d'accés (tant pel administrador com pels diferents gestors de continguts que puguin haver-hi) amb la diferència de que aquí es precisarà introduir un codi de seguretat addicional. Això es fa perquè cap robot virtual pugui accedir al sistema i esborrar-ho tot. Un robot en principi no sap llegir una imatge. Tanmateix si s'escriu malament més de 3 cops un dels tres camps, el sistema bloquejarà la IP del client durant 15 minuts. Aquests valors (temps,intents) es poden modificar des de l'arxiu "panell/constants.inc.php".

NOTA: l'usuari i contrasenya de l'administrador en principi no es pot canviar, si malgrat tot es precisa modificar-los, això s'hauria de fer mitjançant phpMyAdmin (es recorda que la contrasenya s'hauria d'introduir encriptada en md5). Per defecte els valors són:

Usuari	Contrasenya
quamarket	Quamarket85

Pantalla principal del panell de control:



En aquesta plana es mostra el panell de control i les diferents opcions que tenim. Al mig hi tenim un menú en forma d'estrella on poder accedir al contingut del sistema, a les estadístiques de visites de l'aplicació web, a l'administració dels "Gestors de continguts" i una opció per a sortir (de forma segura) del panell. També es pot sortir clicant el petit icona situat abaix a la dreta de la pantalla.

Si l'usuari connectat no fos l'administrador sinó un gestor de contingut aleshores sols veuria el requadre Continguts i Sortir.

Si entrem a la primera plana se'ns presentarà un altra menú amb les següents opcions:



Un gestor de continguts sols veuria el primer requadre

El primer requadre ens dona l'opció d'afegir, modificar i/o eliminar productes del sistema. El segon requadre en canvi ens permet gestionar el que son les categories dels productes, i per tant, el menú que trobem just entrar al sistema. En l'últim requadre en canvi hi trobem la gestió dels usuaris de l'aplicació.

Afegir producte

Si accedim a l'enllaç "Afegir productes" ens apareix la següent pantalla:

Ompli els següents camps i premi Enviar per salvar els canvis:

The screenshot shows a web form for adding a product. It contains the following fields and controls:

- Descripció:** A text input field.
- Preu super A:** A numeric input field with a value of 0.00 and a Euro symbol (€).
- Preu super B:** A numeric input field with a value of 0.00 and a Euro symbol (€).
- Preu super C:** A numeric input field with a value of 0.00 and a Euro symbol (€).
- Familia:** A dropdown menu with the text "--Seleccioni familia--" and a downward arrow.
- Arxiu:** A text input field followed by a button labeled "Examinar...".
- Paraules clau:** A text input field.
- Enviar:** A yellow button with the text "Enviar" located at the bottom right of the form.

Cliqui [aquí](#) per llistar els productes ja inserits

Com es veu, hi ha un formulari que cal omplir correctament abans d'enviar-lo. S'ha d'inserir mínim un preu per producte, una família i una descripció. El

desplegable “família” com ja s'ha dit s'omple des de un altre lloc. Les paraules claus son importants d'introduir si es vol que el cercador cerqui mes fi.

Cal també notar que abaix a l'esquerre hi ha un petit enllaç per a veure un llistat dels productes que ja es tenen inserits fins al moment. Si el cliquem podem observar com s'obre una nova finestra amb els productes disponibles (paginats):

Id	Descripció	PreuA	PreuB	PreuC	Família	Paraules clau
20	Aceite de oliva sabor KOIPE, garrafa 3 l...	8.85	8.5	9	Olis	oli, aceite, oliva, koi...
9	Aceite refinado de girasol - botella de ...	135	12	2	Olis	oli, aceite, girasol, r...
2	Beguda refrescant de Cola...	152	148	16	Cola	beguda, refresc, cola, ...
14	Café molido descafeinado BAQUÉ, paquete ...	2.31	2.1	2.5	Cafès	cafe, descafeinat, baq...
13	Café molido descafeinado mezcla BONKA, p...	2.17	2.2	2.1	Cafès	cafe, descafeinat, bon...
17	Cafe soluble natural NESCAFÉ, caja 10 so...	0.95	1.05	0.9	Cafès	cafe, soluble, natural...
16	Café soluble natural NESCAFÉ, frasco 200...	4.65	4.6	4.6	Cafès	cafe, soluble, natural...
18	Insecticida RAID, moscas y mosquitos, aer...	3.63	3.5	3.63	Aerosol	insecticida, raid, mos...
19	Insecticida RAID, moscas y mosquitos, aer...	3.2	3.3	3.4	Aerosol	insecticida, raid, mos...
22	Ketchup PRIMA, envase 375 g...	1.11	1.05	1.12	Salses	ketchup, tomaquet, pri...

« 1 2 »

14 resultat(s) trobat(s)

Modificar producte:

Si s'hagués accedit en canvi a l'enllaç de modificar producte, el que apareixerà primerament seria una pantalla amb un llistat dels productes actuals com aquesta:

Seleccioni l' **Id** del producte a modificar:

Id	Descripció	PreuA	PreuB	PreuC	Família
20	Aceite de oliva sabor KOIPE, garrafa 3 litros...	8.85	8.5	9	Olis
9	Aceite refinado de girasol - botella de 1 L...	135	12	2	Olis
2	Beguda refrescant de Cola...	152	148	16	Cola
14	Café molido descafeinado BAQUÉ, paquete 250 g...	2.31	2.1	2.5	Cafès
13	Café molido descafeinado mezcla BONKA, paquete 250...	2.17	2.2	2.1	Cafès
17	Cafe soluble natural NESCAFÉ, caja 10 sobres...	0.95	1.05	0.9	Cafès
16	Café soluble natural NESCAFÉ, frasco 200 g...	4.65	4.6	4.6	Cafès
18	Insecticida RAID, moscas y mosquitos, aerosol 750ml...	3.63	3.5	3.63	Aerosol
19	Insecticida RAID, moscas y mosquitos, aerosol 750ml...	3.2	3.3	3.4	Aerosol
22	Ketchup PRIMA, envase 375 g...	1.11	1.05	1.12	Salses

« 1 2 »

14 resultat(s) trobat(s)

Un cop seleccionat el producte que es vol modificar torna a aparèixer una plana com la d'inserir un nou producte però amb els camps ja omplerts:

Descripció	<input type="text" value="Cafe soluble natural NESCAFÉ, caja 10 sobres"/>		
Preu super A	<input type="text" value="0.95"/>	€	
Preu super B	<input type="text" value="1.05"/>	€	
Preu super C	<input type="text" value="0.90"/>	€	
Familia	<input type="text" value="Cafès"/>		
Arxiu	<input type="text"/>	<input type="button" value="Examinar..."/>	Foto actual: 350165.jpg
Paraules clau	<input type="text" value="cafe,soluble,natural,nescafe,sobres"/>		
			<input type="button" value="Enviar"/>

Tots els camps son modificables. Si es desitja canviar la fotografia només cal pujar-ne una de nova, aleshores es substituirà la actual foto. Si en canvi es desitja eliminar la actual sols cal clicar la creu en vermell que hi ha a la dreta d'on diu Foto actual.

Eliminar producte

Si es pretén donar de baixar un producte, caldria clicar sobre l'enllaç Elimina producte del primer requadre. Apareixeria de nou un llistat semblant al de modificar però amb una icona en forma de creu al costat de cada producte. Al clicar sobre un en concret l'eliminem (es demanarà confirmació):

Id	Descripció	PreuA	PreuB	PreuC	Familia	Paraules clau	Oferta
✗20	Aceite de oliva sabor KOIPE, garrafa 3 litros...	8.85	8.5	9	Olis	oli, aceite, oliva, koi...	No
✗9	Aceite refinado de girasol - botella de 1 L...	1.35	1.2	2	Olis	oli, aceite, girasol, r...	No
✗2	Beguda refrescant de Cola...	1.52	1.48	1.6	Cola	beguda, refresc, cola, ...	No
✗14	Café molido descafeinado BAQUÉ, paquete 250 g...	2.31	2.1	2.5	Cafès	cafe, descafeinat, baq...	No
✗13	Café molido descafeinado mezcla BONKA, paquete 250...	2.17	2.2	2.1	Cafès	cafe, descafeinat, bon...	No
✗17	Cafe soluble natural NESCAFÉ, caja 10 sobres...	0.95	1.05	0.9	Cafès	cafe, soluble, natural...	No
✗16	Café soluble natural NESCAFÉ, frasco 200 g...	4.65	4.6	4.6	Cafès	cafe, soluble, natural...	No
✗23	D' ací d' uxò aiàäü "" " i'j'` ++ ` ç...	1	2	1	Cafès	d'ací, d'uxò, d'açañò...	No
✗18	Insecticida RAID, moscas y mosquitos, aerosol 750ml...	3.63	3.5	3.63	Aerosol	insecticida, raid, mos...	No
✗19	Insecticida RAID, moscas y mosquitos, aerosol 750ml...	3.2	3.3	3.4	Aerosol	insecticida, raid, mos...	No

« 1 2 »

[Eliminar tot](#)

També es pot veure en la imatge que hi ha un enllaç “Eliminar tot”. Aquest enllaç s'ha d'usar amb cura ja que ens elimina tot producte de l'aplicació de cop.

NOTA: A l'eliminar un producte, aquest s'eliminarà també de les llistes dels usuaris. D'on no s'eliminarà en canvi serà dels historials de compra dels usuaris ja que allí no s'hi guarda cap relació directe amb el producte (sols text).

Gestió de categories

Ja passant al segon requadre d'opcions, allà hi trobem la gestió de categories com s'observa a continuació:

Nom categoria:

Subcategoria de: -- Seleccione --

Enviar

Categories amb les seves respectives subcategories:

Afeitat (Higiene)	X
Aigües (Begudes)	X
Alcohol (Begudes)	X
Alimentació (Mascotes)	X
Ambientadors (Drogueria)	X
Aperitius	X
Basar	X
Begudes	X
Bombetes (Basar)	X
Cafès (Esmorzars)	X
Capil·lar (Higiene)	X
Carnisseria (Frescs)	X
Congelats	X
Conserves i cuina	X
Corporal (Higiene)	X
Detergents (Drogueria)	X
Drogueria	X
Esmorzars	X
Estris de neteja (Basar)	X
<u>Mopes</u>	X
<u>Plumeros</u>	X
<u>Juquets</u>	X
<u>Begudes</u>	X

Si no hi hagués cap categoria entrada, sols podríem entrar-ne una nova categoria pare. Un cop fet això ja podríem donar d'alta noves categories pare o categories filles tot seleccionant del desplegable la categoria que li faria de “pare”.





Si una de les categories es vol eliminar sols fa falta clicar la icona de la seva dreta. Es demanarà confirmació. Aquesta opció el que farà serà eliminar la categoria

junt amb els productes que estan relacionats amb ella (llistes i historial inclosos). Es aquí on entra en acció les relacions creades en la base de dades i l'eliminació en "cascada". Si el que s'elimina es una categoria pare aleshores totes les seves categories filles s'esborraran també junt amb els productes que pengin d'elles.

Modificar usuari web

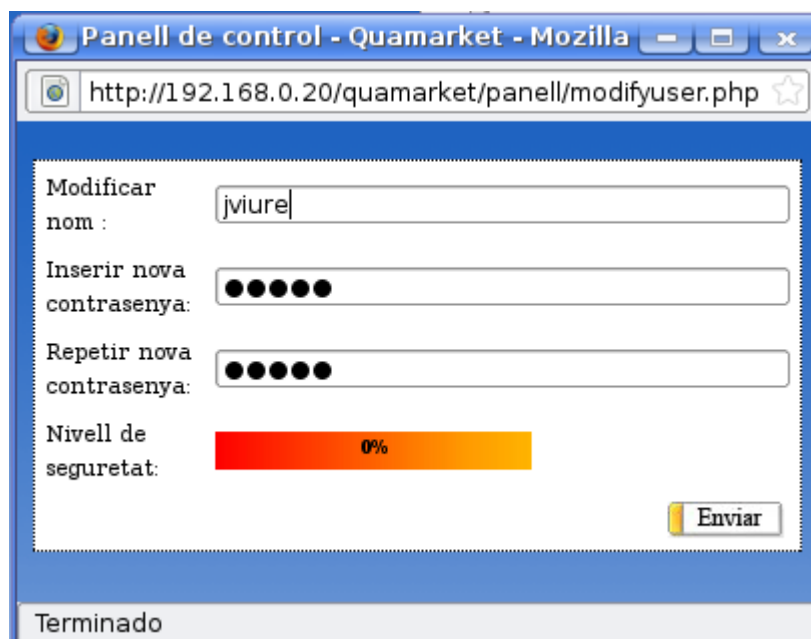
Si el que es desitja es modificar el nom d'usuari o la contrasenya d'un usuari del sistema caldrà accedir a l'enllaç del tercer bloc del menú de continguts. La pantalla que es vorà serà la següent:

A continuació es llisten els diferents usuaris registrats del sistema. Si desitja modificar la contrasenya d'algún cliqui a l'icone groga del seu lateral:

ID	Nom	
1	jviure	
6	jotauve	
12	joan	
13	ronaldo	



Es llisten els diferents usuaris existents junt amb un icona que al ésser clicat s'obre la següent finestra:



Aquest formulari ve a ésser com el primer que es mostra al donar-se d'alta en el sistema. Al clicar Enviar es salvaran els canvis si tot es correcte, es tancarà la finestra i s'actualitzarà automàticament la finestra pare (el llistat).

Eliminar usuari

De la mateixa manera que s'eliminen productes, amb usuaris passa el mateix, la finestra es la següent:

A continuació es llisten els diferents gestors de contingut del sistema. Si desitja modificar la contrasenya d'algún cliqui a l'icone groga del seu lateral:

ID	Nom	
1	jviure	✖
6	jotauve	✖
12	joan	✖
13	ronaldo	✖

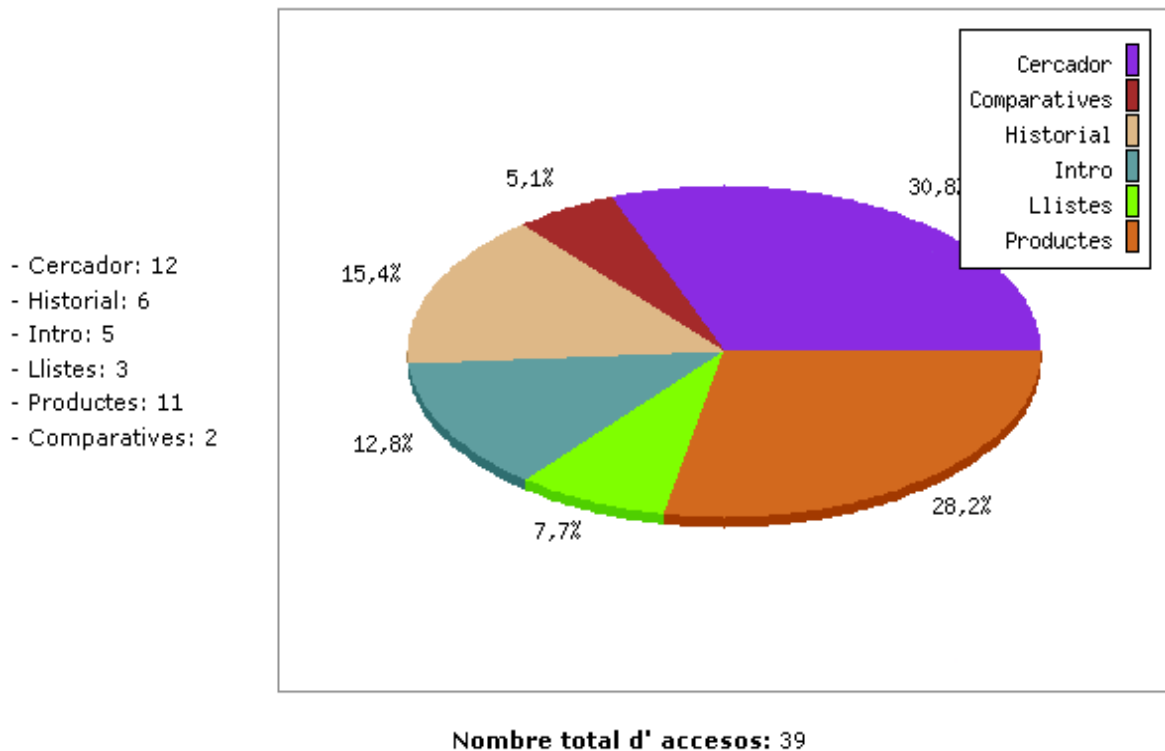


Es demanarà confirmació abans de procedir a la seva baixa. Junt amb l'eliminació de l'usuari s'esborraran les seves llistes personalitzades i els seus historials de compra.

Estadístiques del sistema

A mode informatiu s'ha inserit en la pantalla principal del panell de control una opció per a poder veure les estadístiques d'accessos al sistema. Si s'hi accedeix es pot trobar la següent gràfica:

A continuació es mostra un llistat de cada una de las planes que conforma la Web junt amb les seves visites:



A l'esquerre es mostra numèricament els accessos de cada part del sistema i a la dreta es mostra el mateix però de manera gràfica. Amb aquest gràfic un es pot fer una idea de quins son els apartats més usats del sistema i veure si fa falta millorar-los o no en funció del seu ús general.

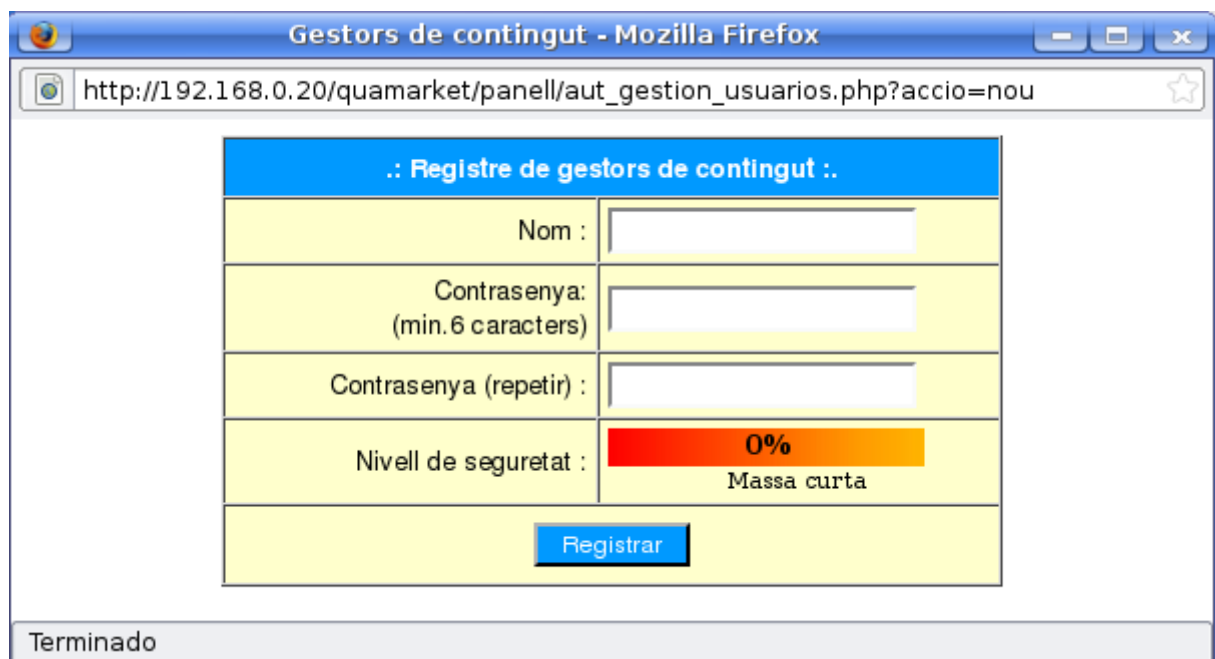
Alta,Baixa,Modificació de Gestors de continguts

Es interessant poder disposar (si mai s'escala el sistema) de diferents persones que puguin anar omplint la base de dades del sistema (apart de l'administrador). A continuació es mostra la plana que apareix un cop clicat l'enllaç "Gestors":



En aquesta plana es podrà registrar un nou gestor o esborrar-ne un d'existent. També es pot veure qui es l'administrador però aquest no es pot eliminar ja que es únic.

Alhora de registrar-ne un de nou es presenta un formulari com el que segueix:



La idea és la mateixa que la ja mostrada alhora de donar d'alta un usuari web. Un cop enviat el formulari es llistarà el nou gestor en la pantalla anterior.

NOTA: Si l'aplicació no percep cap interacció durant 1440 segons, es procedirà a tallar la sessió a l'usuari. Aquesta restricció de seguretat es troba per defecte en la configuració del servidor Apache, es pot modificar mitjançant l'arxiu *php.ini*.