

Gestió de compres d'una petita empresa.

Nom Estudiant: Joan Muniesa Rubio
ETIG

Nom Consultor: Jordi Ceballos Villach

Data Lliurament: 11/06/2007

2.- Resum.

L'empresa "Llibres i multimèdia, S.L" ens ha encarregat una petita aplicació que porti un control dels articles i proveïdors que té actualment. Aquesta empresa es dedica bàsicament a vendre llibres, articles de papereria, elements multimèdia i joguines.

L'objectiu d'aquesta aplicació bàsicament és tenir unes dades mínimes d'aquests ítems per tal d'elaborar un llistat Excel que serveixi de punt de partida per a fer-lo evolucionar segons les circumstàncies de l'empresa en cada moment.

Bàsicament, l'aplicació permetrà:

- Realitzar altes, baixes, modificacions i consultes de treballadors de l'empresa.
- Dels treballadors donats d'alta, podrà afegir, modificar, consultar o eliminar perfils d'entrada a l'aplicació.
- Podrà gestionar els diferents proveïdors de cada article que ven.
- Gestionar cada concepte que intervé en l'aplicació en forma de manteniment.
- Llistar cada gestió de l'aplicació segons diferents criteris de selecció.

És per això que aquesta aplicació no ha de tenir totes les dades dels treballadors, per exemple, ja que aquestes estan guardades en la base de dades central de l'empresa.

A més a més, l'aplicació permetrà configurar un usuari amb un o més perfils. Encara que un usuari tingui diversos perfils per entrar a l'aplicació, només ens interessarà guardar l'acció realitzada per l'usuari i no amb quin perfil ha realitzat aquesta acció.

Aquesta aplicació l'hem anomenat "THOT Cultura", nom que ens ha semblat adient per ser curt i de fàcil pronunciació. A més cal indicar com a curiositat, que Thot va ser el Déu de l'escriptura, les biblioteques i la llengua en l'antic Egipte, per això ens ha semblat correcte utilitzar-lo per a una aplicació d'una empresa on un dels seus principals articles son els llibres.

El projecte THOT Cultura, consisteix en el control de compres de l'empresa així com en el control i registre dels diferents proveïdors amb els que opera.

Aquest control es duu a terme registrant totes les compres que es realitzen de cada una de les quatre àrees amb les que opera aquesta empresa.

A més del registre de compres de cada una de les àrees citades, també hi ha un mòdul de gestió només accessible per l'administrador que permet definir donar accés als diferents treballadors de l'empresa segons una sèrie de perfils ja preestablerts i no configurables.

Tota aquesta gestió es resumeix en un llistat en format Excel on d'una forma ràpida es pot veure tota la gestió de l'aplicació.

Aquest projecte s'ha desenvolupat de forma íntegra en la plataforma .NET 2.0 a través de l'aplicació de Microsoft Visual Studio 2005, concretament, s'ha utilitzat el llenguatge de programació c# i la base de dades utilitzada per a desar tota la informació és també de Microsoft i es tracta de SQL Server 2005.

A banda de tot això, també s'ha utilitzat una tecnologia que s'està expenent molt ràpidament i que dona més sensació d'aplicació NO web evitant la recàrrega innecessària de les diferents pantalles de l'aplicació, estem parlant de la tecnologia AJAX (anomenat ATLAS ja que utilitzem també la versió de Microsoft).

Per a la generació del llistat Excel hem utilitzat la suite Microsoft Office 2003.

3.- Índex de continguts i índex de figures.

2.- Resum.....	2
3.- Índex de continguts i índex de figures.....	4
4.- Capítol 1. Introducció.	
4.1.- Justificació del PFC i context en el qual es desenvolupa: punt de partida i aportació del PFC.....	6
4.2.- Objectius del PFC.	7
4.3.- Enfocament i mètode seguit.....	8
4.4.- Planificació del projecte.	10
4.5.- Productes obtinguts	12
4.6.- Breu descripció dels altres capítols de la memòria. .	13
5.- Capítol 2. Anàlisi	
5.1.- Diagrames de casos d’ús.	15
5.2.- Diagrama de classes del model conceptual	27
6.- Capítol 3. Disseny	
6.1.- Diagrama de l’arquitectura software	29
6.1.1.- Introducció a AJAX	30
6.2.- Diagrama de l’arquitectura hardware	33
6.3.- Diagrama de classes de disseny mostrant la separació per capes i amb una descripció dels principals atributs i mètodes de la classe.	34
6.4.- Disseny de la interfície d’usuari.	41
6.5.- Disseny de la base de dades.	51
7.- Conclusions	56
8.- Línies de desenvolupament futur.....	57
9.- Glossari.....	58
10.- Bibliografia.....	59

Índex de Figures.

Figura 1. Funcionalitats agrupades per perfil.....	6
Figura 2. Fases del projecte.....	8
Figura 3. Documents i dates de lliurament.....	10
Figura 4. Detall de tasques concretes i temporització.....	11
Figura 5. Diagrama de Gantt	12
Figura 6. Productes obtinguts	14
Figura 7. Cas d’us general.....	15
Figura 8. Cas d’us d’articles.....	15
Figura 9. Casos d’us de l’aplicació.....	16
Figura 10. Model conceptual.....	28
Figura 11. Aplicació web clàssica vs aplicació amb tecnologia AJAX.....	31
Figura 12. Component ValidatorCalloutExtender en acció.....	32
Figura 13. Estructura hardware amb IIS i SQL Server separats.	33
Figura 14. Estructura hardware amb IIS i SQL Server junts.	33
Figura 15. Estructura en capes	35
Figura 16. Diagrama de classes.	36
Figura 17. Fluxe de pantalles.....	43
Figura 18. Pantalla de login.	45

Figura 19. Pantalla de accés no autoritzat.....	45
Figura 20. Selecció de perfil.....	46
Figura 21. Pantalla d’inici.....	46
Figura 22. Pantalla de selecció d’articles.....	47
Figura 23. Manteniment d’articles.....	47
Figura 24. Selecció de treballadors.....	48
Figura 25. Manteniment de treballadors.....	48
Figura 26. Manteniment de logins.....	48
Figura 27. Manteniment de tipus d’article.....	49
Figura 28. Selecció de proveïdors.....	49
Figura 29. Manteniment de proveïdors.....	49
Figura 30. Paràmetres del llistat Excel.....	50
Figura 31. Llistat Excel generat.....	50
Figura 32. Diagrama E/R.....	51
Figura 33. Camps de les taules de la base de dades.....	56

4.- Introducció.

4.1.- Justificació del TFC i context en el qual es desenvolupa: punt de partida i aportació del TFC

El que es persegueix amb aquest projecte és completar el software de l'empresa client realitzant una petita aplicació que registri les compres dels diferents articles amb els que treballa, agrupats en quatre grups (de moment, ampliables en un futur), aquests grups són:

- Papereria (bolígrafs, paquets de fulls, llàpissos, grapadores ...)
- Llibres (de text, d'aventura, infantils ...)
- Productes multimèdia (consumibles per impressores, DVDs, CDs, ratolins ...)
- Joguines (grans i petites, per a qualsevol edat)

Amb aquesta informació i la dels proveïdors que ofereixen els seus productes a l'empresa, es genera un llistat en format MS Excel on es pot veure un resum de l'import gastat per proveïdor, per exemple.

En resum, tota la informació introduïda es sintetitza en un únic full de càlcul amb la informació que necessita el client.

Al ser una aplicació molt senzilla i al treballar amb conceptes que ja coneixen els treballadors, no és necessari atura l'empresa per a realitzar una exhaustiva formació, simplement entrant a l'aplicació (un cop l'administrador ha donat accés) és fàcil fer-se amb ella.

L'aplicació consta de 3 grups de funcionalitats que són accessibles pels usuaris registrats segons el perfil donat per l'administrador:

Aquestes funcionalitats agrupades per perfil són les següents:

Perfil	Funcionalitat
Administrador	<ul style="list-style-type: none">- Gestió d'articles- Manteniment de taules mestres- Gestió del llistat Excel
Visualitzador	<ul style="list-style-type: none">- Gestió d'articles (només consulta)- Manteniment de taules mestres (només consulta)
Operador	<ul style="list-style-type: none">- Gestió d'articles

Figura 1. Funcionalitats agrupades per perfil

Per a la realització del projecte, s’ha utilitzat Visual Studio 2005 de Microsoft que treballa amb el Framework 2.0.

Segons la Wikipedia:

“El Framework o “marc de treball”, constitueix la base de la plataforma .NET i avarca la infraestructura sobre la qual es reuneixen un conjunt de llenguatges, eines i serveis que simplifiquen el desenvolupament d’aplicacions en entorn d’execució distribuït.”

Nosaltres hem utilitzat íntegrament el llenguatge de programació C#, que és el “llenguatge insígnia” de .NET. La seva sintaxi deriva de C i C++ i presenta força similituds amb el llenguatge Java.

A part d’aquest llenguatge de programació, hem utilitzat altres tecnologies, com són ADO .NET per a l’accés a dades, ASP .NET i tractament de dades estàtiques amb XML.

En quant a l’aportació d’aquest TFC és mostrar un projecte que pot ser utilitzat inicialment com a base per a qualsevol petit negoci de compra-venda de qualsevol tipus de productes, ja que els tipus de producte són parametrizables. Caldria, a partir d’aquesta base, realitzar tasques de desenvolupament per adaptar-lo a les necessitats concretes i específiques de l’empresa en qüestió.

4.2.- Objectius del TFC

L’objectiu principal és el familiaritzar-se amb una tecnologia i, de forma més concreta, amb un llenguatge de programació que està començant a donar fort en el mercat actual. A banda d’aquest aspecte tant general, podem concretar més indicant els objectius a més baix nivell, aquests són els següents:

- Aprendre la filosofia de .NET. Veure el llenguatge c# en el conjunt de .NET
- Comprendre i aplicar la tecnologia AJAX de Microsoft.
 - o Conèixer la problemàtica de les aplicacions webs (postbacks) i l’evolució històrica que ha sofert aquesta tecnologia:
 - Remote Scripting
 - Callbacks ...
- Desenvolupar aplicacions web amb tecnologia Microsoft (ASP)
- Comprendre els mètodes i la integració que ens ofereix Microsoft entre les seves aplicacions.
 - o Utilització de MS Excel des de les pàgines ASP.
- Conèixer SQL Server 2005
 - o Programació amb T-SQL (Stored Procedures)
- Conèixer i aplicar el disseny en les pàgines segons regles que marca W3C.

4.3.- Enfocament i mètode seguit

El projecte que tenim entre mans, s’ha enfocat des de la perspectiva del cicle de vida típic de qualsevol projecte software (també s’anomena en cascada) però el mètode seguit no ha estat aquest ben bé ja que hi ha parts que no s’han arribat a realitzar (ni es realitzaran) ja que l’etapa de manteniment no està definida aquí. A més, el cicle de vida en cascada no permet la revisió de passos anteriors (tornar enrere) i en el nostre cas, s’han realitzat petites modificacions i retocs de cada part segons avançava el desenvolupament del projecte, independentment de la fase en la que ens trobéssim.

S’ha aprofitat la forma de treballar de la UOC, en PACs per utilitzar cada una de les tres en que consta el projecte per a fer entrega d’una part del projecte o fase, d’aquesta manera és més fàcil ajustar-nos als plaços d’entrega marcats i garantir l’entrega del producte final en la data pactada.

Cal dir també, que s’han elaborat uns documents que serveixen com a “guia d’instal·lació” del producte. Aquestes guies acostumen a ser molt breus però suposen una ajuda inestimable per a l’usuari final ja que s’indica, a banda del producte en si, com descarregar per a la seva instal·lació i com instal·lar component utilitzat en l’aplicació i que, moltes vegades, des de el punt de vista de l’usuari final, és difícil conèixer la seva existència i per tant, no aconseguir “arrencar” l’aplicació.

A continuació mostrarem les diferents fases utilitzades així com la PAC amb la que han estat associades.

Com es podrà veure posteriorment, cada fase serà definida i explicada detalladament:

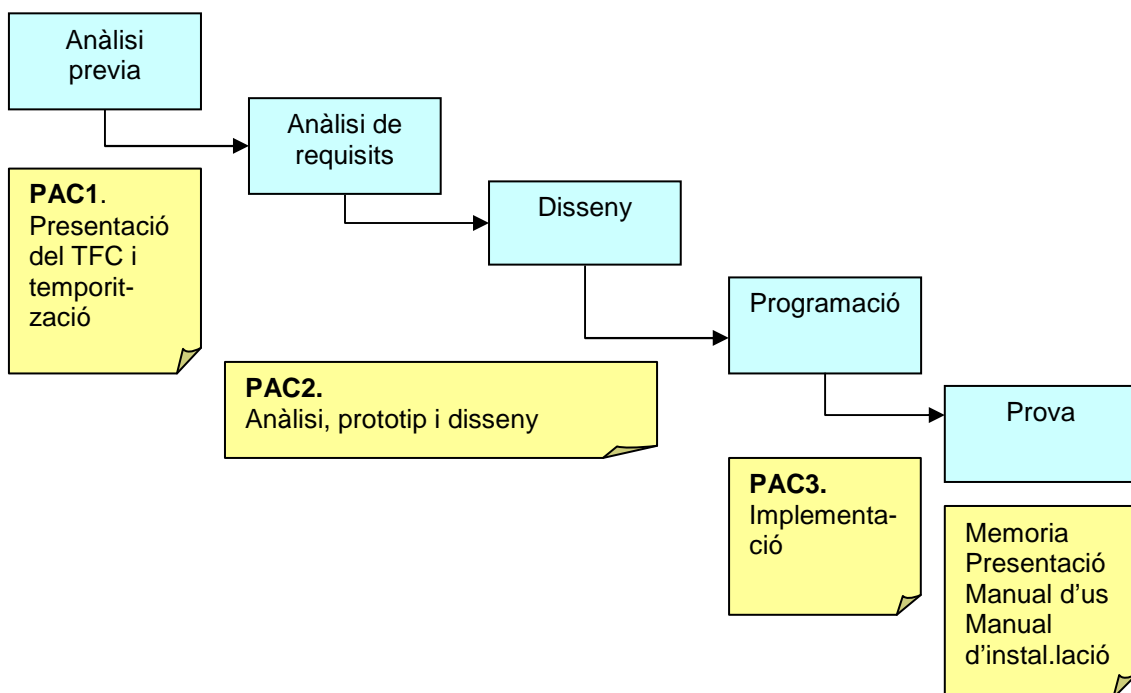


Figura 2. Fases del projecte

Com podem veure en la figura superior, hem indicat cada fase en forma de cascada i, en groc, el contingut de cada lliurament parcial (PAC) com a parts d'un tot que és el TFC.

4.4.- Planificació del projecte.

La planificació seguida és la mateixa que la elaborada al inici de l’assignatura coincidint amb la PAC1.

En la figura següent es pot veure una taula amb les dates clau acompanyada d’informació del treballa realitzat.

Data de lliurament	Document/fita	Descripció
13/03/2007	PAC1. Especificació del sistema	Document on es parla del projecte escollit així com la seva planificació temporal.
09/04/2007	PAC2. Anàlisi. Especificació de requisits. Especificació de disseny.	Definició més precisa del projecte a desenvolupar. Es concreta el disseny E/R de la base de dades a utilitzar i el diagrama estàtic amb UML de les classes a utilitzar. També es lliure una maqueta o prototip de l’aplicació per a veure el seu disseny gràfic, funcionalitats així com transicions de pantalles.
28/05/2007	PAC3. Implementació	Implementació de l’aplicació i joc de proves. Obtenim un software complet i funcional
11/06/2007 Lliurament Final	Memòria i presentació virtual.	La memòria és el present documento n es resumeix i on agrupem tota la informació detallada en les fases anteriors. La presentació virtual és un resum de la memòria. A més a més, es lliuren dos documents que són el manual de instal·lació i el manual d’usuari.

Figura 3. Documents i dates de lliurament.

En la següent figura, podem veure de forma més detallada, la temporització de cada tasca concreta.

	i	Nombre de tarea	Duración	Comienzo	Fin	Pr
1		Lectura documents .NET	77 días?	vie 23/02/07	lun 11/06/07	
2		Instal·lació software Microsoft	3 días?	lun 05/03/07	mié 07/03/07	
3		Pràctica AJAX	6 días?	vie 09/03/07	dom 18/03/07	
4		Definir Pla de Treball	3 días?	vie 09/03/07	mar 13/03/07	
5		Creació del site (IIS) i visibilitat exterior	13 días	mié 14/03/07	sáb 31/03/07	
6		Taula de Login, permisos i perfils i SP d'accés	5 días	mié 14/03/07	mar 20/03/07	
7		Pantalla de Login	3 días	mié 21/03/07	vie 23/03/07	
8		Definició d'estils de l'aplicació (css)	3 días?	lun 26/03/07	mié 28/03/07	
9		Creació XMLs d'opcions de menús	2 días?	jue 29/03/07	vie 30/03/07	
10		Aplicar drets als perfils segons logins y xml de perfils	4 días?	lun 02/04/07	jue 05/04/07	
11		Manteniment de llibres	3 días	jue 05/04/07	lun 09/04/07	
12		PAC 2 Entrega	1 día?	lun 09/04/07	lun 09/04/07	
13		Manteniment de papereria	3 días	mar 10/04/07	jue 12/04/07	
14		Manteniment de articles multimèdia	3 días?	vie 13/04/07	mar 17/04/07	
15		Manteniment de Joguines	3 días?	mié 18/04/07	vie 20/04/07	
16		Test dels fet fins ara	5 días?	lun 23/04/07	vie 27/04/07	
17		Creació de llistats de cada manteniment	5 días?	lun 30/04/07	vie 04/05/07	
18		Creació de llistat Excel	7 días	lun 07/05/07	mar 15/05/07	
19		Memòria i documentació	26 días?	lun 07/05/07	lun 11/06/07	
20		Test General	19 días?	mié 16/05/07	lun 11/06/07	
21		Preparació d'enviament: scripts sql	4 días?	vie 18/05/07	mié 23/05/07	

Figura 4. Detall de tasques concretes i temporització.

Una visió més global de tota l’extensió del projecte, la podem trobar en el Diagrama de Gantt que mostrem a continuació.

Tant en el detall de tasques concretes de la figura 4 com en el diagrama de Gantt següent (figura 5), veurem que hi han algunes tasques que es solapen amb altres degut a la necessitat de disposar de més temps (independent de la feina de desenvolupament) pera aconseguir la fita desitjada, algunes d’aquestes tasques poden ser la formació en les tecnologies que utilitzarem en el projecte i la creació del site en IIS i aconseguir una visibilitat des de l’exterior.

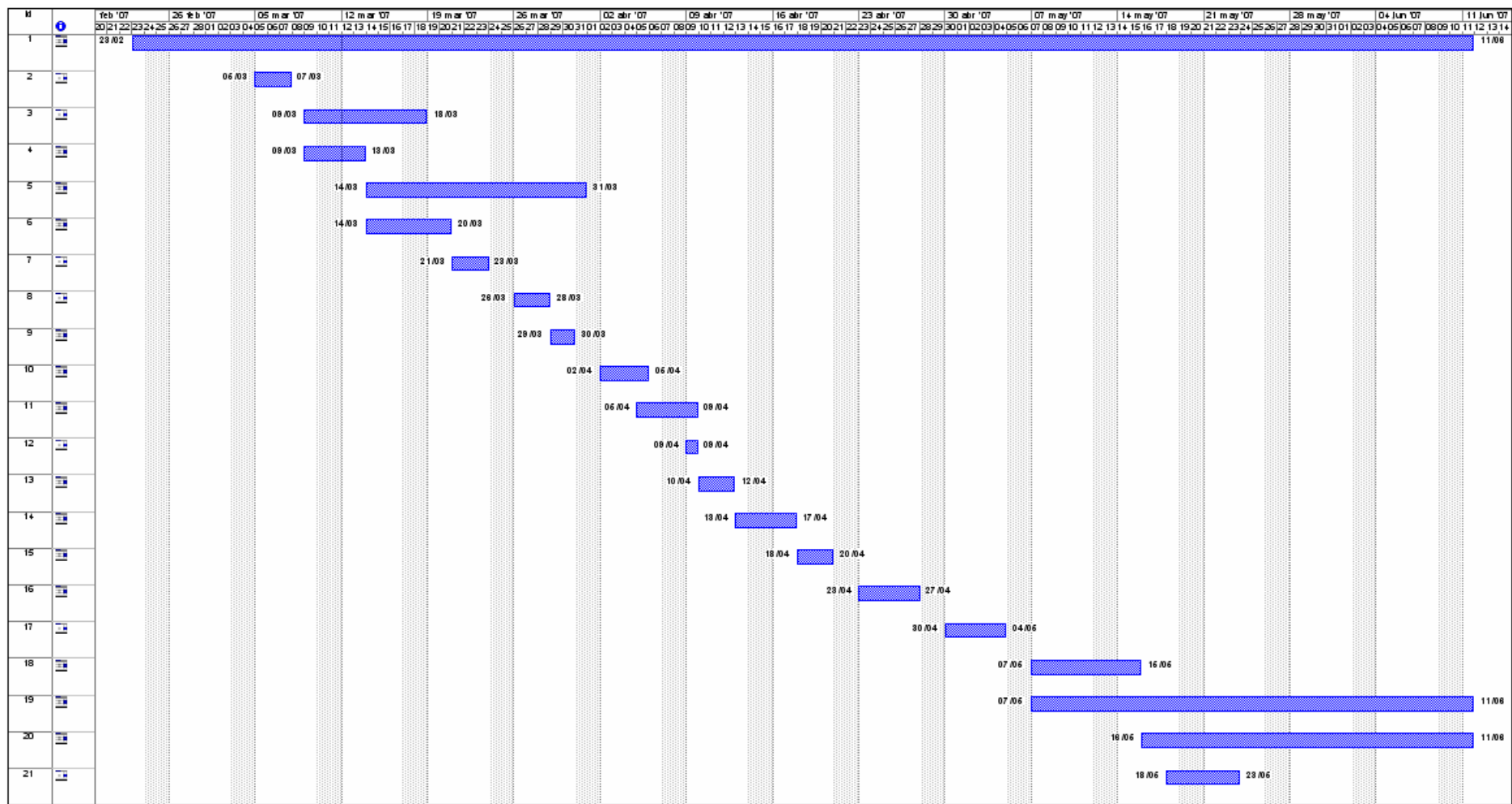


Figura 5. Diagrama de Gantt

4.5.- Productes obtinguts.

Els productes obtinguts en la realització d'aquest projecte coincideixen una mica amb els punts descrits anteriorment de cada entrega parcial o PAC del projecte, a continuació es comentaran una mica.

Producte	Descripció
Pla de treball	<p>Plantejament de la temàtica escollida així com la tecnologia a utilitzar. També es defineixen els requeriments funcionals i tècnics.</p> <p>Es present la planificació temporal de cada tasca/subtasca dins del projecte per tal d'assolir les dates d'entrega, tot això queda reflectit en el Diagrama de Gantt adjuntat en aquesta fase.</p>
Anàlisi	<p>Es detalla cada part del projecte i com funciona, també es postren els requeriments funcionals de l'aplicació per mitjà de l'inclusió dels diferents casos d'us i la descripció de cada un d'ells.</p> <p>També es decideix el disseny que tindrà la base de dades i les diferents classes que formaran l'aplicació.</p>
Prototip	<p>Es mostra una maqueta de l'aplicació final amb dades reals "escrites" (sense accés a la base de dades ni utilització de classes) que donen una visió global de com serà el producte un cop finalitzat.</p> <p>D'aquesta forma es pot veure de una forma no tant abstracte els dissenys indicats en la part d'anàlisi.</p>
Implementació	<p>Correspon al desenvolupament "pur i dur" aplicant tot el que s'ha indicat en la part d'anàlisi. Aquest desenvolupament correspon tant a l'aplicació web com a la lògica implementada en els Stored Procedures de la base de dades.</p>
Memòria del projecte	Correspon a aquest document.
Presentació del projecte	És una document en format PowerPoint que dona una visió

	general de totes les parts implicades en el projecte en format més resumit, donant més força a les parts més importants i que puguin generar una mica de confusió.
Manual d'instal·lació	Document on s'explica cada detalladament en passos per tal de realitzar una instal·lació correcta, l'aplicació en si i els diferents components que la formen alguns d'ells de tercers.
Manual de funcionament	Manual detallat que serveix de guia al l'usuari final, per tal de realitzar un bon us de l'aplicació i no tenir dubte de cap apartat, opció de menú i, en definitiva, funcionalitat.

Figura 6. Productes obtinguts

4.6.- Descripció dels altres capítols de la memòria.

A continuació farem un breu resum de la resta de capítols que constitueixen la memòria.

Capítol 2: Anàlisi. En aquest capítol trobarem una descripció més acurada dels diferents aspectes funcionals amb la descripció del diagrama de casos i la descripció de cada cas d'us que hi ha a l'aplicació, també trobem, en aquest capítol, el disseny de classes i una explicació de cada un dels mètodes utilitzats

Capítol 3: Disseny. Aquí explicarem l'arquitectura utilitzada a nivell de software i la de hardware, explicarem també, el disseny de classes i una explicació de cada un dels mètodes utilitzats, mostrarem també el disseny de la interfície d'usuari amb diferents imatges de pantalles que formen l'aplicació.

Finalitzarem el capítol mostrant el disseny de la base de dades.

Capítol 3: Conclusions. Explicarem quines conclusions hem tret amb l'elaboració d'aquest projecte així com de la tecnologia emprada.

Capítol 4: Línies de desenvolupament futur. Si volguéssim estendre i ampliar les funcionalitats d'aquesta aplicació, què fariem? Aquesta pregunta té resposta en aquest capítol.

5.- Anàlisi.

5.1.- Diagrama de casos d'us i explicació de cada cas d'us.

En una primera aproximació, identifiquem els actors que interactuen amb el sistema. Per millorar la comprensió i major simplicitat i coherència, només considerarem l'operador o gestor com la resta de perfils NO administradors que interactuaran amb el sistema de venda de llibres.

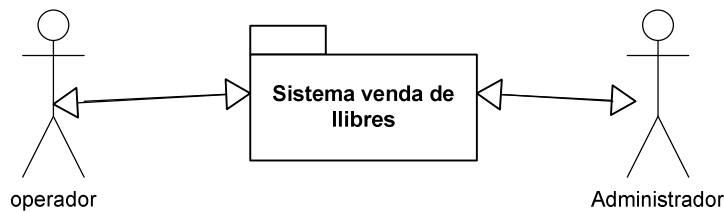


Figura 7. Cas d'us general

A continuació anem a realitzar un aclariment. En el diagrama general veurem que parlem d'articles, quan en realitat un article, com hem dit abans, pot ser un producte de papereria, multimèdia, una joguina, etc ...

Hem d'imaginar, per exemple, el cas d'us "alta d'article" de la següent forma:

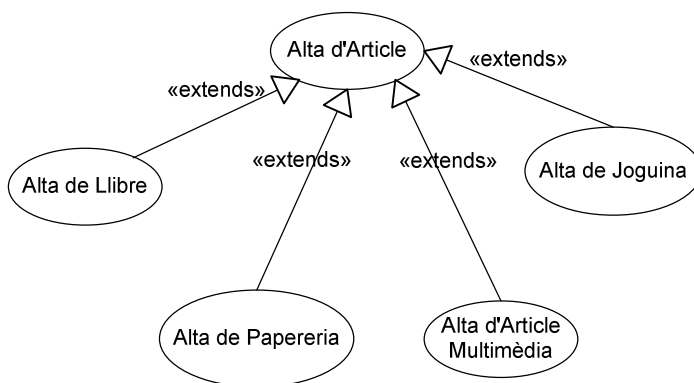


Figura 8. Cas d'us d'articles

Però que, un cop més, per raons de simplicitat, simplement parlarem de "alta d'article".



Figura 9. Casos d'us de l'aplicació.

A continuació, detallarem cada cas d'ús per a la seva millor comprensió.

Cas d'ús Identificar-se

Resum de la funcionalitat: inicia una sessió en l'aplicació d'un treballador de l'empresa

Actors: **Administrador, Operador**

Casos d'ús relacionats: Tots.

Precondició: L'empleat existeix i té un perfil com a mínim relacionat

Postcondició: -

Procés normal principal:

- 1.- El sistema demana el nom d'usuari i el password del treballador
- 2.- L'**empleat** introdueix les dades indicades al pas 1
- 3.- El sistema verifica les dades.

Cas d'ús Alta d'article

Resum de la funcionalitat: Afegeix un article de qualsevol tipus a la base de dades.

Actors: **Administrador, Operador**

Casos d'ús relacionats: Modificació d'article, consulta d'article, eliminació d'article

Precondició: L'usuari està identificat i l'article no existeix

Postcondició: S'enregistra un nou article.

Procés normal principal:

- 1.- S'introdueixen les dades del nou article indicant també el tipus que és.
- 2.- El sistema verifica les dades.
- 3.- El sistema enregistra les dades.

Cas d'ús Modificació d'article

Resum de la funcionalitat: Modifica un article de qualsevol tipus existent a la base de dades.

Actors: **Administrador, Operador**

Casos d'ús relacionats: Consulta d'article

Precondició: L'usuari està identificat i l'article existeix

Postcondició: S'enregistren les modificacions realitzades sobre l'article existent.

Procés normal principal:

- 1.- Es fa una cerca per trobar l'article que es vol modificar,
- 2.- Es modifiquen les dades de l'article.
- 3.- El sistema verifica les dades.
- 4.- El sistema enregistra les dades.

Cas d'ús Consulta d'article

Resum de la funcionalitat: Consulta un article de qualsevol tipus a la base de dades.

Actors: **Administrador, Operador**

Casos d'ús relacionats: Modificació de l'article, eliminació de l'article.

Precondició: L'usuari està identificat

Postcondició: Es troba l'article cercat o no.

Procés normal principal:

- 1.- Es fa una cerca per mirar de trobar l'article sempre i quan la data d'eliminació no estigui informada.
- 2.- El sistema mostra per pantalla un llistat sobre les coincidències trobades segons els criteris de cerca indicats.
- 3.- L'usuari selecciona el registre que vol de la llista mostrada.
- 4.- el sistema retorna la informació mostrant una fitxa amb totes les dades de l'article.

Alternatives de procés i excepcions:

2a.- El sistema no troba l'article i no mostra res.

2b.- Finalitza el cas d'us

Cas d'ús Eliminació de l'article

Resum de la funcionalitat: No s'elimina físicament de la base de dades. Es modifica la data d'eliminació i l'usuari que ha provocat l'eliminació d'un article de qualsevol tipus a la base de dades.

Actors: **Administrador, Operador**

Casos d'ús relacionats: Consulta d'article

Precondició: L'usuari està identificat i l'article existeix.

Postcondició: S'informen els camps de data d'eliminació i usuari d'eliminació de l'article i passa a no ser tingut en compte.

Procés normal principal:

- 1.- Es fa una cerca per mirar de trobar l'article (suposem que es troba)
- 2.- El sistema mostra per pantalla un llistat sobre les coincidències trobades segons els criteris de cerca indicats.
- 3.- L'usuari selecciona el registre que vol de la llista mostrada.
- 4.- el sistema retorna la informació mostrant una fitxa amb totes les dades de l'article.
- 5.- Des de la fitxa es selecciona l'eliminació de l'article.
- 6.- El sistema modifica les dades.

Cas d'ús Impressió Llistat

Resum de la funcionalitat: Consulta un article de qualsevol tipus a la base de dades.

Actors: **Administrador, Operador**

Casos d'ús relacionats: Consulta de l'article.

Precondició: L'usuari està identificat i ha tret una llista en pantalla dels registres que s'adapten als criteris de selecció indicats.

Postcondició: Es treu un llistat per impressora.

Procés normal principal:

- 1.- Es fa una cerca per mirar de trobar una llista d'articles sempre i quan la data d'eliminació no estigui informada.
- 2.- El sistema mostra per pantalla un llistat sobre les coincidències trobades segons els criteris de cerca indicats.
- 3.- L'usuari indica que vol treure el llistat per impressora.
- 4.- El sistema retorna la informació mostrant la informació de cada registre per impressora..

Cas d'ús Alta de treballador

Resum de la funcionalitat: Afegeix un treballador a la base de dades.

Actors: **Administrador**

Casos d'ús relacionats: Modificació de treballador, consulta de treballador, eliminació de treballador.

Precondició: L'usuari està identificat i el treballador no existeix.

Postcondició: S'enregistra un nou treballador a la base de dades.

Procés normal principal:

- 1.- S'introdueixen les dades del nou treballador.
- 2.- El sistema verifica les dades.
- 3.- El sistema enregistra les dades.

Cas d'ús Modificació de Treballador

Resum de la funcionalitat: Modifica un treballador existent a la base de dades.

Actors: **Administrador.**

Casos d'ús relacionats: Consulta de Treballador

Precondició: L'usuari està identificat i el treballador existeix.

Postcondició: S'enregistren les modificacions realitzades sobre el treballador existent.

Procés normal principal:

- 1.- Es fa una cerca per trobar el treballador que es vol modificar,
- 2.- Es modifiquen les dades del treballador.
- 3.- El sistema verifica les dades.
- 4.- El sistema enregistra les dades.

Cas d'ús Consulta de Treballador

Resum de la funcionalitat: Consulta un treballador a la base de dades.

Actors: **Administrador**

Casos d'ús relacionats: Modificació del treballador, eliminació del treballador.

Precondició: L'usuari està identificat

Postcondició: Es troba el treballador cercat o no.

Procés normal principal:

- 1.- Es fa una cerca per mirar de trobar el treballador sempre i quan la data d'eliminació no estigui informada.
- 2.- El sistema mostra per pantalla un llistat sobre les coincidències trobades segons els criteris de cerca indicats.
- 3.- L'usuari selecciona el registre que vol de la llista mostrada.
- 4.- El sistema retorna la informació mostrant una fitxa amb totes les dades del treballador.

Alternatives de procés i excepcions:

2a.- El sistema no troba el treballador i no mostra res.

2b.- Finalitza el cas d'ús

Cas d'ús Eliminació del treballador

Resum de la funcionalitat: El treballador no serà eliminat físicament de la base de dades, simplement modificarem la data d'eliminació i l'usuari que ha provocat l'eliminació d'un treballador a la base de dades.

Actors: **Administrador**

Casos d'ús relacionats: Consulta de Treballador

Precondició: L'usuari està identificat i el treballador existeix en la base de dades.

Postcondició: Es modifiquen els camps de data d'eliminació i usuari d'eliminació del treballador i passa a no ser tingut en compte en cap altre cas d'ús.

Procés normal principal:

- 1.- Es fa una cerca per mirar de trobar el treballador (suposem que es troba)
- 2.- El sistema mostra per pantalla un llistat sobre les coincidències trobades segons els criteris de cerca indicats.
- 3.- L'usuari selecciona el registre que vol de la llista mostrada.
- 4.- el sistema retorna la informació mostrant una fitxa amb totes les dades del treballador.
- 5.- Des de la fitxa es selecciona l'eliminació del treballador.
- 6.- El sistema modifica les dades.

Cas d'ús Alta de Login

Resum de la funcionalitat: Afegeix a un treballador existent en la base de dades un o més perfils d'entrada a l'aplicació. En funció del perfil d'entrada podrà realitzar unes accions o unes altres.

Actors: **Administrador**

Casos d'ús relacionats: Modificació de treballador, consulta de treballador, Alta de treballador, consulta de Login, modificació de Login.

Precondició: L'usuari està identificat i el treballador existeix.

Postcondició: S'enregistra un nou login a la base de dades.

Procés normal principal:

- 1.- S'introdueixen les dades del login seleccionant un treballador de la base de dades.
- 2.- El sistema verifica les dades.
- 3.- El sistema enregistra les dades.

Cas d'ús Modificació de Login

Resum de la funcionalitat: Modifica les dades d'entrada a l'aplicació d'un treballador existent a la base de dades amb un login també existent.

Actors: **Administrador.**

Casos d'ús relacionats: Consulta de Treballador, Consulta login.

Precondició: L'usuari està identificat, el treballador existeix i el login d'aquest treballador existeix

Postcondició: S'enregistren les modificacions realitzades sobre el login existent.

Procés normal principal:

- 1.- Es fa una cerca per trobar el login del treballador que es vol modificar,
- 2.- Es modifiquen les dades del seu login.
- 3.- El sistema verifica les dades.
- 4.- El sistema enregistra les dades.

Cas d'ús Consulta de Login

Resum de la funcionalitat: Consulta un login d'un treballador a la base de dades.

Actors: **Administrador**

Casos d'ús relacionats: Modificació del treballador, eliminació del treballador, modificació de login eliminació de login.

Precondició: L'usuari està identificat

Postcondició: Es troba el login segons la cerca indicada o no.

Procés normal principal:

- 1.- Es fa una cerca per mirar de trobar el login del treballador sempre i quan la data d'eliminació no estigui informada, tant del treballador com del seu login.
- 2.- El sistema mostra per pantalla un llistat sobre les coincidències trobades segons els criteris de cerca indicats.
- 3.- L'usuari selecciona el registre que vol de la llista mostrada.

4.- El sistema retorna la informació mostrant una fitxa amb totes les dades del perfil del treballador.

Alternatives de procés i excepcions:

2a.- El sistema no troba el treballador i/o perfil i no mostra res.

2b.- Finalitza el cas d'ús

Cas d'ús Eliminació del Login

Resum de la funcionalitat: El login no serà eliminat físicament de la base de dades, simplement modificarem la data d'eliminació i l'usuari que ha provocat l'eliminació d'un login a la base de dades.

Actors: **Administrador**

Casos d'ús relacionats: Consulta de Treballador, consulta de login.

Precondició: L'usuari està identificat, el treballador existeix en la base de dades així com el login indicat.

Postcondició: Es modifiquen els camps de data d'eliminació i usuari d'eliminació del login.

Procés normal principal:

- 1.- Es fa una cerca per mirar de trobar el login d'un treballador concret o tots els logins d'un treballador concret (suposem que es troba)
- 2.- El sistema mostra per pantalla un llistat sobre les coincidències trobades segons els criteris de cerca indicats.
- 3.- L'usuari selecciona el registre que vol de la llista mostrada.
- 4.- el sistema retorna la informació mostrant una fitxa amb totes les dades del login.
- 5.- Des de la fitxa es selecciona l'eliminació del login.
- 6.- El sistema modifica les dades.

Cas d'ús Generació Llistat especial XML

Resum de la funcionalitat: Es realitza una cerca per tal d'extreure unes dades mínimes amb les que es generarà un llistat per ser interpretat per MS Excel.

Actors: **Administrador**

Casos d'ús relacionats: Consulta d'article, Consulta de tipus d'article, Consulta de Proveïdor, Consulta de Treballador.

Precondició: L'usuari està identificat, hi ha articles, proveïdors ... en la base de dades.

Postcondició: S'obté un llistat en format MS Excel pel seu tractament posterior amb aquesta eina ofimàtica..

Procés normal principal:

- 1.- Es fa una cerca de diversos camps per tal d'obtenir un llistat
- 2.- El sistema mostra per pantalla un llistat sobre les coincidències trobades segons els criteris de cerca indicats.
- 3.- L'usuari indica que vol extreure la informació en format MS Excel
- 4.- El sistema retorna la informació obrint el resultat en MS Excel.

Cas d'ús Alta de Tipus d'articles

Resum de la funcionalitat: És un manteniment que permet crear tipus d'articles per tal de classificar els articles. Afegeix un tipus d'article en la base de dades.

Actors: **Administrador**

Casos d'ús relacionats: Modificació de tipus d'article, consulta de tipus d'article, eliminació de tipus d'article .

Precondició: L'usuari està identificat i el tipus d'article no existeix.

Postcondició: S'enregistra un nou tipus d'article a la base de dades.

Procés normal principal:

- 1.- S'introdueixen les dades del login seleccionant un treballador de la base de dades.
- 2.- El sistema verifica les dades.
- 3.- El sistema enregistra les dades.

Cas d'ús Modificació de Tipus d'articles

Resum de la funcionalitat: Modifica les dades d'un tipus d'article existent.

Actors: **Administrador**.

Casos d'ús relacionats: Consulta de Tipus d'article.

Precondició: L'usuari està identificat i el tipus d'article existeix.

Postcondició: S'enregistren les modificacions realitzades sobre el tipus d'article existent.

Procés normal principal:

- 1.- Es mostren tots els tipus d'article (no es realitza cerca ja que no hi hauran gaires) en un llistat.
- 2.- Es selecciona el registre que es vol modificar.
- 3.- Es modifiquen les dades del tipus d'article seleccionat.
- 4.- El sistema verifica les dades.
- 5.- El sistema enregistra les dades.

Cas d'ús Consulta de Tipus d'article

Resum de la funcionalitat: Consulta un tipus d'article a la base de dades.

Actors: **Administrador**

Casos d'ús relacionats: Modificació del tipus d'article, eliminació del tipus d'article. Precondició: L'usuari està identificat

Postcondició: Es troba el tipus d'article entre tots els que hi ha.

Procés normal principal:

- 1.- El sistema mostra per pantalla un llistat de tots els tipus d'article no eliminats.

Cas d'ús Eliminació de Tipus d'article

Resum de la funcionalitat: El tipus d'article no serà eliminat físicament de la base de dades, simplement modificarem la data d'eliminació i l'usuari que ha provocat l'eliminació d'un tipus d'article a la base de dades.

Actors: **Administrador**

Casos d'ús relacionats: Consulta de tipus d'article.

Precondició: L'usuari està identificat, el tipus d'article existeix en la base de dades.

Postcondició: Es modifiquen els camps de data d'eliminació i usuari d'eliminació del login.

Procés normal principal:

- 1.- El sistema mostra per pantalla un llistat amb tots els tipus d'article no eliminats
- 2.- L'usuari indica que vol eliminar el registre que seleccioni de la llista mostrada.
- 3.- El sistema modifica les dades.

Cas d'ús Alta de Proveïdor

Resum de la funcionalitat: Afegeix un proveïdor a la base de dades.

Actors: **Administrador**

Casos d'ús relacionats: Modificació de proveïdor, consulta de proveïdor, eliminació de proveïdor.

Precondició: L'usuari està identificat al sistema i el proveïdor no existeix.

Postcondició: S'enregistra un nou proveïdor a la base de dades.

Procés normal principal:

- 1.- S'introdueixen les dades del nou proveïdor.
- 2.- El sistema verifica les dades.
- 3.- El sistema enregistra les dades.

Cas d'ús Modificació de Proveïdor

Resum de la funcionalitat: Modifica un proveïdor existent a la base de dades.

Actors: **Administrador**.

Casos d'ús relacionats: Consulta de Proveïdor

Precondició: L'usuari està identificat i el proveïdor existeix.

Postcondició: S'enregistren les modificacions realitzades sobre el proveïdor existent.

Procés normal principal:

- 1.- Es fa una cerca per trobar el proveïdor que es vol modificar,
- 2.- Es modifiquen les dades del proveïdor.
- 3.- El sistema verifica les dades.
- 4.- El sistema enregistra les dades.

Cas d'ús Consulta de Proveïdor

Resum de la funcionalitat: Consulta un proveïdor a la base de dades.

Actors: **Administrador**

Casos d'ús relacionats: Modificació del proveïdor, eliminació del proveïdor.

Precondició: L'usuari està identificat

Postcondició: Es troba el proveïdor cercat o no.

Procés normal principal:

- 1.- Es fa una cerca per mirar de trobar el proveïdor sempre i quan la data d'eliminació no estigui informada.
- 2.- El sistema mostra per pantalla un llistat sobre les coincidències trobades segons els criteris de cerca indicats.
- 3.- L'usuari selecciona el registre que vol de la llista mostrada.
- 4.- El sistema retorna la informació mostrant una fitxa amb totes les dades del proveïdor.

Alternatives de procés i excepcions:

2a.- El sistema no troba el proveïdor i no mostra res.

2b.- Finalitza el cas d'us

Cas d'ús Eliminació del proveïdor

Resum de la funcionalitat: El proveïdor no serà eliminat físicament de la base de dades, simplement modificarem la data d'eliminació i l'usuari que ha provocat l'eliminació d'un proveïdor a la base de dades.

Actors: **Administrador**

Casos d'ús relacionats: Consulta de Proveïdor

Precondició: L'usuari està identificat i el proveïdor existeix en la base de dades.

Postcondició: Es modifiquen els camps de data d'eliminació i usuari d'eliminació del proveïdor i a efectes pràctics, és com si no existís.

Procés normal principal:

- 1.- Es fa una cerca per mirar de trobar el proveïdor (suposem que es troba)
- 2.- El sistema mostra per pantalla un llistat sobre les coincidències trobades segons els criteris de cerca indicats.
- 3.- L'usuari selecciona el registre que vol de la llista mostrada.
- 4.- el sistema retorna la informació mostrant una fitxa amb totes les dades del proveïdor.
- 5.- Des de la fitxa es selecciona l'eliminació del proveïdor.
- 6.- El sistema modifica les dades.

5.2.- Diagrama de classes del model conceptual.

En el model conceptual, tenim una representació dels conceptes del mon real, no de components de software, sense entrar en gaires detalls.

El diagrama de classes del model conceptual, no és més que el conjunt de diagrames d'estructura estàtic amb classes, atributs i associacions però no operacions o mètodes.

A continuació, mostrem el diagrama de classes del model conceptual:

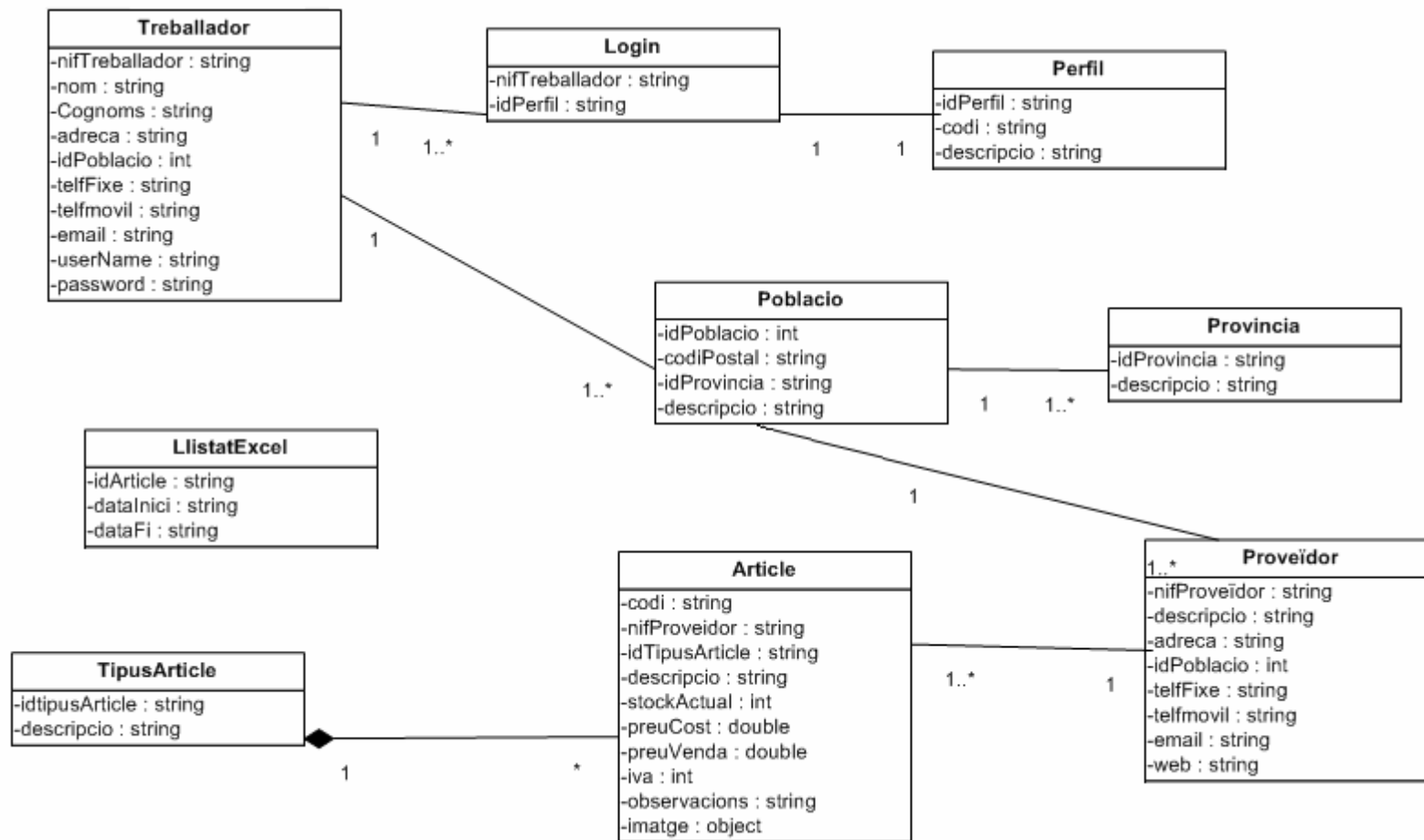


Figura 10. Model conceptual.

6.- Disseny.

6.1.- Diagrama de l'arquitectura software.

Em escollit realitzar aquest projecte amb tecnologia Microsoft perquè per la nostra experiència ens ha mostrat major fiabilitat, no obstant, cal dir que sabem de l'existència d'altres tecnologies amb prestacions similars i inversions econòmiques molt inferiors però encara i així hem preferit utilitzar Microsoft.

Aquesta aplicació es desenvoluparà, doncs, en .NET 2005, per ser la darrera de les tecnologies de desenvolupament per la que està apostant fort Microsoft, concretament el desenvolupament es realitzarà en el llenguatge c#.

Les diferents opcions de menú estaran indicades en tants fitxers XML com perfils hi hagin, d'aquesta forma, simplement editant el fitxer XML corresponent a cada perfil podrem incorporar o treure opcions de menú a aquests perfils.

Per a la generació del llistat de gestió s'ha pensat realitzar-ho per mitjà de MS Excel.

S'ha pensat en que l'aplicació estigui dissenyada per a un accés en entorn web ja que només amb el navegador d'Internet serà suficient i no tindrem que instal·lar res en les màquines clients i això permetrà en un futur proper fer créixer aquesta aplicació per tal que la puguin utilitzar els comercials amb els seus portàtils, per exemple, per realitzar les comandes ens les seves visites als clients i, en qualsevol moment, realitzar una replicació de dades, per mitjà, per exemple, de la tecnologia de "publicador-suscriptor".

Per això necessitarem configurar l'aplicació en un servidor que tingui instal·lat Internet Information Services.

Al tractar-se de tecnologia Microsoft, la base de dades escollida serà Microsoft SQL Server 2005, l'accés a cada una de les taules es realitzarà a través de stored procedures.

A banda de tot això, s'utilitzarà una altra tecnologia anomenada AJAX, existeixen en el mercat diferents solucions que la implementen, però un cop més utilitzarem la que ens ofereix Microsoft.

6.1.1.- Introducció a AJAX.

Per tal de comprendre la necessitat d'utilitzar (i utilitzar correctament) aquesta nova tecnologia, farem una petita incursió en l'evolució d'aquesta vessant tecnològica.

En la programació en entorns web, sempre teníem l'inconvenient de realitzar la recàrrega de tota la pàgina en la que estàvem sempre que volíem realitzar una crida per a retornar un valor o omplir una llista en funció d'una tria realitzada, etc.

Aquest recàrrega era inevitable i alhora innecessària creant malestar en l'usuari i en l'empresa ja que, innecessàriament, es generava un tràfic de dades en la xarxa de l'empresa.

Aquest tràfic i aquest malestar general dels usuaris que utilitzaven aplicacions basades en web es veia incrementat per connexions lentes com eren, fins fa ben poc, les connexions a través de mòdems.

Es van realitzar intents de solucionar aquest problema com va ser el remote scripting i fins va més poc els callbacks ja en .NET.

Aquesta nova tecnologia anomenada AJAX (Asynchronous Javascript and XML) no és ben bé una nova tecnologia com a tal, si no la suma de diverses tecnologies existents treballant de forma conjunta que dona lloc a una nova forma de programar aplicacions web interactives.

Aquesta evolució de DHTML s'ha anomenat web 2.0. Per a fer-ho Ajax utilitza XHTML i CSS per a formatar la informació, DOM (document Object Model) per a interactuar i per a visualitzar dinàmicament la informació es recolza en XML i JavaScript per actualitzar les dades sense necessitat de refrescar la pàgina.

Permet realitzar crides al servidor asíncrones i recolzant-se en tecnologia ja existent com és el XML evitant la càrrega total de la pàgina i només centrant-se en aquell tros, en aquell component o objecte que es vol actualitzar. D'aquesta manera, treballar en entorn web es pot començar a semblar a treballar en un aplicatiu desenvolupat, per exemple, en Windows forms i instal·lat en la màquina client.

Per exemple, en un procés normal una aplicació web recarrega la mateixa pàgina o va d'una pàgina a una altra perquè ha d'enviar peticions al servidor web i la s'ha de processar la petició i enviar el resultat final (que no és més que la pàgina). En tot això AJAX fa el mateix però amb la diferència que ho fa en background és a dir, que realitza tot el procés de comunicació amb el servidor sense que l'usuari pugui veure-ho i un cop terminat amb èxit només ha d'afegir el resultat del procés a la pàgina actual sense tenir que recarregar (com hem dit) la pàgina, es a dir, actualitza "porcions" de la pàgina sense recarregar de nou tot el contingut del document en el navegador.

Podem trobar excel·lents exemple on s'utilitza AJAX en GoogleMaps, GMail, etc ...

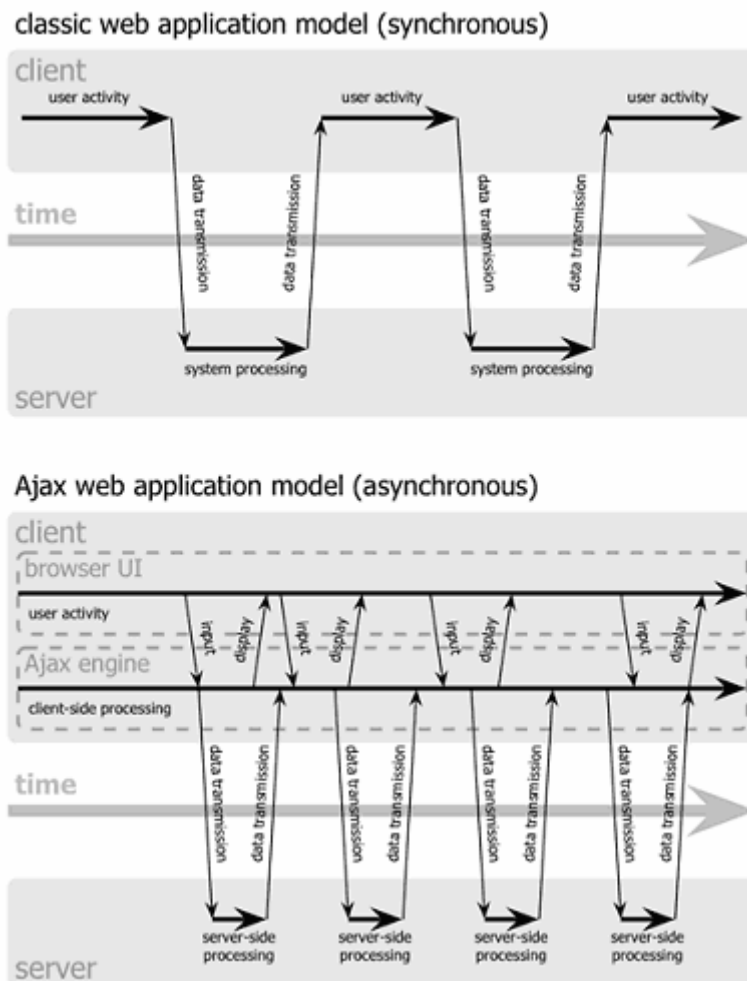


Figura 11. Aplicació web clàssica vs aplicació amb tecnologia AJAX

En el projecte THOT Cultura, podem veure aquesta tecnologia AJAX en acció , concretament amb el Microsoft Ajax Toolkit, que és un conjunt de components que extenen el funcionament d'alguns controls estàndard.

Exactament, en aquest projecte, hem utilitzat el component "cascadingDropDown" que extén la funcionalitat d'una llista desplegable o dropDown fent que s'ompli de forma asíncrona, per mitjà de crides a mètodes implementats en un webservice i, condicionant aquesta selecció de dades per a omplir el dropDown, a una selecció realitzada abans en un altre dropDown, per això rep el nom de "en cascada" (cascading).

En el nostre cas concret, cada cop que vulguem recuperar una població primer seleccionarem una província de totes les províncies carregades en una llista desplegable de províncies. Un cop seleccionada una, automàticament i sense tenir que fer postbacks (recàrregues de pàgina) innecessaris, l'aplicació tindrà

que fer una crida asíncrona a un procediment emmagatzemat que llençarà una consulta per a retornar la descripció de la població i actualitzar la llista de poblacions de la pantalla. Tot això sense necessitat de recarregar tota la pàgina per a propagar el canvi i minimitzant el tràfic en xarxa.

Aquest component el podem trobar en els manteniments de treballadors i en de proveïdors.

Un altre component AJAX que em utilitzat en l'aplicació ha estat l'anomenat "ValidatorCalloutExtender", que extén les possibilitats del "requiredFieldValidator" (que és un component estàndard que s'encarrega de vetllar perquè els camps indicats siguin informats ja que són camps "requerits") i oferir un petit "popup" o finestra emergent amb una interfície molt professional indicant la obligatorietat d'informar aquest camp.

Aquesta "finestreta" és bastant personalitzable podent escollir a part del text que es visualitzarà per a informar de l'obligatorietat del camp diferents imatges com poden ser, per exemple i en el nostre cas, una imatge d'exclamació i també una petita imatge a la cantonada superior dreta simulant el botó de "tancar" de qualsevol altre aplicació.

Aquest component el trobem, per exemple, en la pantalla inicial de "login" on, si no indiquem cap nom d'usuari, apareix aquest "popup" informant-nos d'aquesta circumstància.



Figura 12. Component ValidatorCalloutExtender en acció.

Hem de dir, després d'utilitzar aquests components, que l'experiència ha estat increïble en el sentit que et fa pensar la potencia que té AJAX per tal de recarregar els "combos" de població sense notar un mínim de refresc en la pàgina, com si realment estiguéssim davant d'una aplicació "webForms", a banda d'això, també ens fa pensar en la resta de tecnologies que poden estar sense descobrir i el que caldria simplement és combinar altres tecnologies existents, és a dir, que segurament es podrà augmentar i dotar a les aplicacions web o no web de major potència combinant tecnologies ja existents.

6.2.- Diagrama de l'estructura hardware.

Com hem dit, l'aplicació serà en entorn web i amb una base de dades SQL server 2005, tot tecnologia Microsoft.

Encara que la millor arquitectura és la que utilitza dos servidors idèntics tots dos amb IIS i MS SQL Server instal·lat però només en un d'ells correrà la web i en l'altre estarà ubicada la base de dades. És bo que siguin idèntics ja que, en cas de que deixi de funcionar la base de dades o el IIS de cada servidor podem utilitzar la base de dades (prèvia restauració) i el IIS dels servidors contraris.

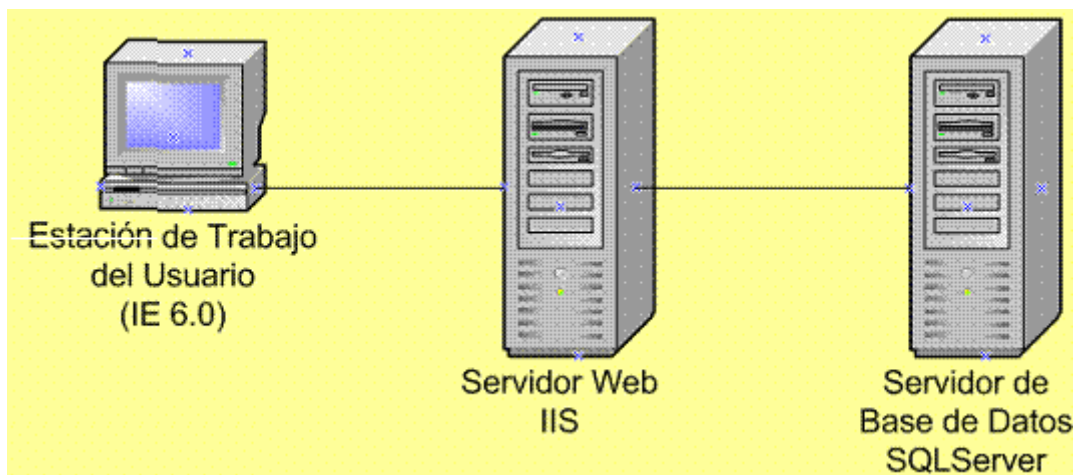


Figura 13. Estructura hardware amb IIS i SQL Server separats.

No obstant, en el nostre cas, i donades les característiques de l'aplicació (aplicació senzilla sense gaires accessos a la base de dades) utilitzarem un únic servidor on conviurà tant la base de dades com IIS, tal i com mostra la següent imatge:

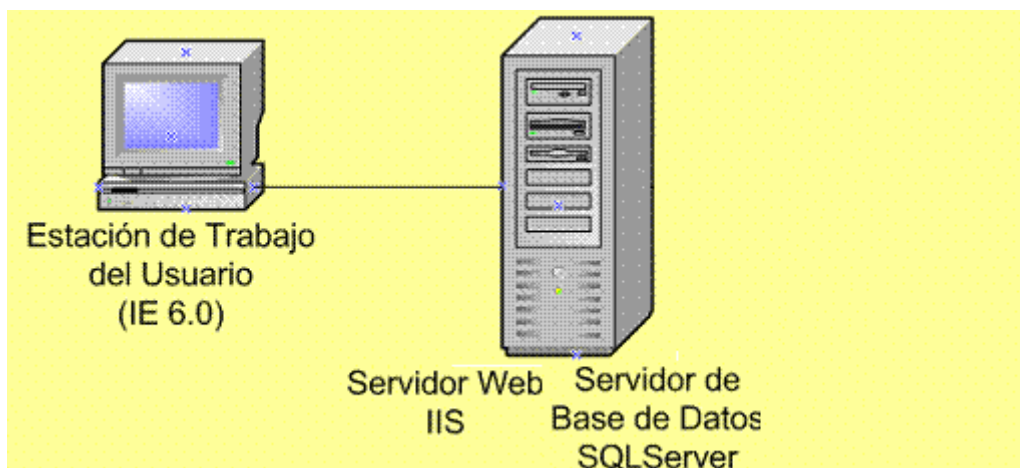


Figura 14. Estructura hardware amb IIS i SQL Server junts.

Les característiques mínimes de la màquina servidora seran:

A nivell de hardware serà suficient amb:

- CPU Intel Pentium 4 a 3.2 Ghz.
- 1 Gb de memòria ram
- Connexió ADSL 1Mb

A nivell de software:

- SO Windows 2003 Server
- IIS 5
- MS SQL Server 2005

Creiem que, donada la concurrència d'usuaris que es preveu serà màquina suficient per a donar Server sense cap tipus de disminució en el rendiment esperat per els usuaris.

En quan als clients donat que estem treballant amb tecnologia Microsoft, recomanem que s'utilitzi el navegador Internet Explorer 6.0 o versions superiors.

6.3.- Diagrama de classes de disseny mostrant la separació per capes i amb una descripció dels principals atributs i mètodes de la classe.

En aquesta aplicació podem trobar diverses capes encara que no segueix un patró definit de forma molt estricte.

És a dir, hi ha una capa de presentació (capa UI) i interacció amb el GUI de l'aplicació i amb l'usuari que "passa" informació a una altra capa que correspon a una sèrie de classes que interactuen directament amb la base de dades com si fos una capa DAL (Data Access Layer o capa d'accés a dades), però com hem dit, no sempre és així, podem trobar vegades que la mateixa capa UI realitzi accessos directament a la base de dades i també implementi lògica de negoci, tot en un.

De forma general, l'estructura en capes de l'aplicació és:

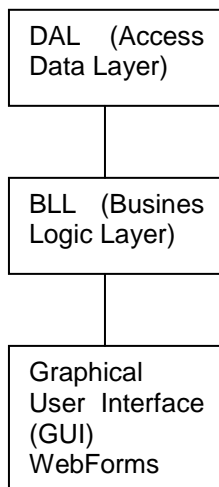


Figura 15 Estructura en capes

L'usuari interactua amb la capa GUI o UI realitzada amb webForms i que correspon a les pantalles de l'aplicació.

Qualsevol acció que fa, provoca que aquesta capa interactuï amb la capa BLL que és on està la lògica de negoci, aquesta, en determinades ocasions, necessitarà accedir a les dades emmagatzemades en la base de dades mitjançant crides a Stored Procedures (o a les taules directament en altres vegades), aquesta acció la fa la capa DAL.

A continuació mostrarem el disseny de classes, comentarem cada mètode i després intentarem realitzar una classificació de les classes en les diferents capes.

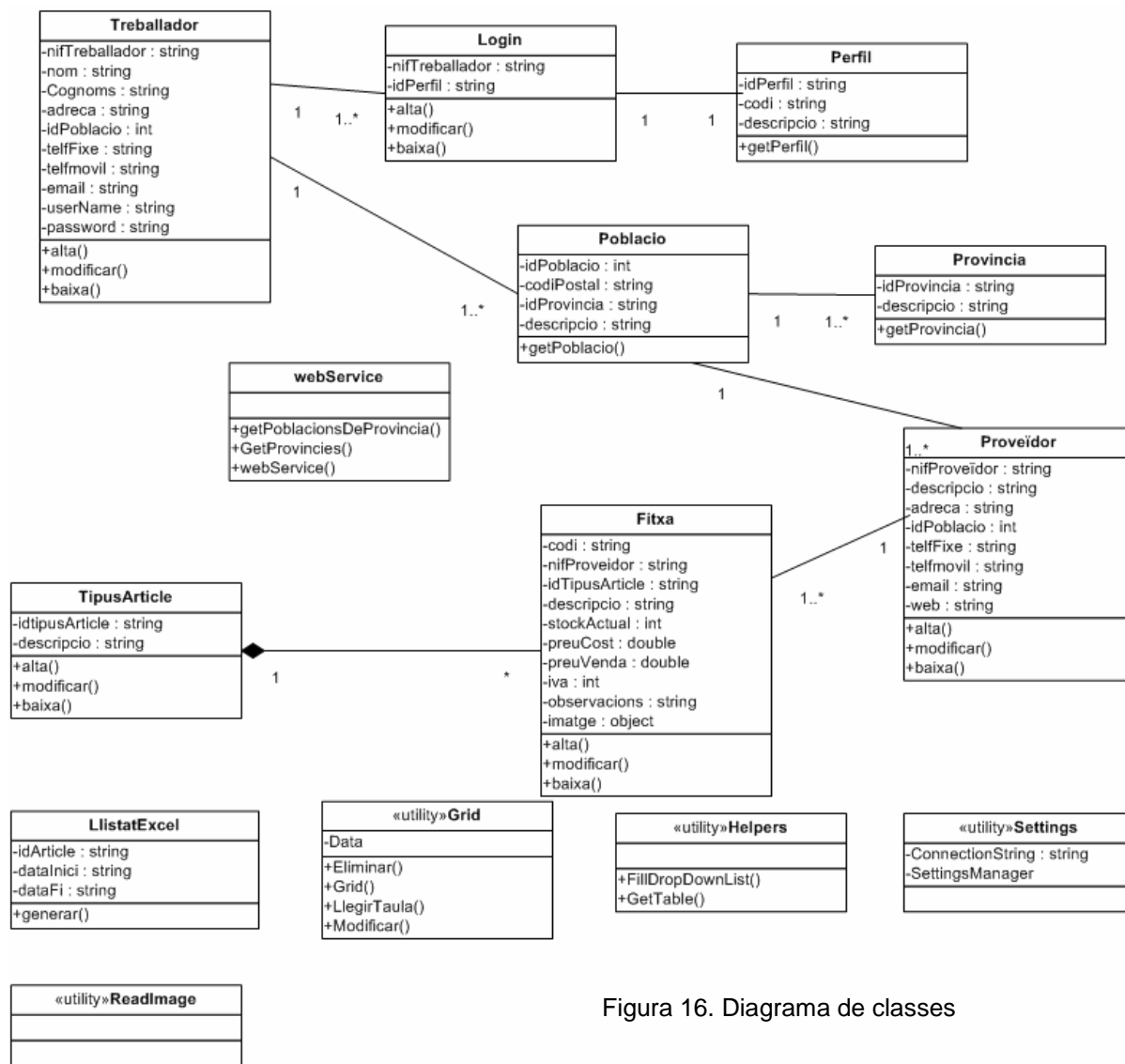


Figura 16. Diagrama de classes

Podem veure tot el conjunt de l'aplicació amb un seguit de classes relacionades i després unes classes aïllades que corresponen a classes d'utilitat (indicades amb l'estereotip "utility") que són classes que no afecten a cap objecte en concret, simplement són això, utilitzats que serveixen per a simplificar determinades tasques.

Per el nom, es pot veure clarament que hi ha una sèrie de classes que estarien dins la capa DAL, i són les següents:

- Classe Grid
- Classe Helpers

Són classes que accedeixen directament a la base de dades a través de consultes (cadena de text on està la sentència SQL o bé el nom del Stored Procedure a executar).

La resta de classes estan emmarcades dins la capa de presentació i alhora realitzen tasques de la capa BLL i, fins i tot, de DAL pel que el límit de cada una està molt difós.

A continuació explicarem cada classe així com cada mètode que la forma:

Classe Treballador:

Aquesta classe representa a un treballador de la empresa que s'ha donat d'alta en l'aplicació per tal de convertir-se en un nou usuari de l'aplicació. És per això que no és necessari guardar totes les seves dades, ja que estaran guardades en la base de dades central de l'empresa sinó unes dades bàsiques que seran les necessàries per la nostra aplicació.

Els mètodes d'aquesta classe són:

- Alta: Permet donar d'alta un treballador en l'aplicació. Aquest mètode cridarà a un objecte de la base de dades que realitzarà l'alta física en la base de dades.
- Modificar: Permet realitzar modificacions sobre un treballador existent en la nostra base de dades i que no s'hagi eliminat, mitjançant la crida a un Stored Procedure de la nostra base de dades.
- Baixa: a través del identificador d'aquest treballador, aquest mètode es comunicarà a un objecte de la base de dades que l'eliminarà físicament.

Classe Login:

Donat un treballador existent i no eliminat, aquesta classe ens permetrà relacionar-lo amb un perfil. Un treballador podrà tenir un o diferents perfils. En funció del perfil, podrà realitzar unes accions o unes altres.

Els mètodes d'aquesta classe són:

- Alta: Permet relacionar un perfil amb un treballador existent. Aquest mètode cridarà a un objecte de la base de dades que realitzarà l'alta física en la base de dades.
- Modificar: Permet canviar el perfil d'un treballador existent en la nostra base de dades, mitjançant la crida a un Stored Procedure de la nostra base de dades.
- Baixa: a través del identificador d'aquest treballador i el seu perfil, aquest mètode es comunicarà a un objecte de la base de dades que eliminarà físicament el registre.

Classe Perfil:

Es gestionaran els accessos als perfils. Recordem que aquesta taula no es mantindrà. Els perfils que hi hauran inicialment seran:

Administrador: Se li permetrà realitzar qualsevol acció en l'aplicació i no tindrà cap restricció.

Visualitzador: De les opcions de menú que tingui activades, només podrà visualitzar les dades després d'haver recuperat registres a través de consultes.

Operador o gestor: no tindrà accés a totes les opcions de menú i tindrà determinades restriccions en quan a modificacions de determinats camps.

Classe Poblacio:

Aquesta classe simplement contindrà les crides per a accedir a la base de dades i recuperar les característiques d'una població. Aquesta classe només tindrà mètodes accessors que retornin la descripció de la població o poblacions en el cas de que aquestes siguin petites i puguin haver diverses en el mateix codi postal. És per això que en l'aplicació, aquests valors es recolliran en una llista desplegable.

Classe Provincia:

Contindrà les crides per a accedir a la base de dades i recuperar les característiques d'una província. Aquesta classe només tindrà mètodes accessors que recuperaran la seva descripció.

Classe TipusArticles:

Permet gestionar els diferents tipus d'articles amb els que treballarà la nostra aplicació.

Cada article ha de pertànyer obligatòriament a un tipus d'article que permeti classificar-ho.

Els mètodes d'aquesta classe són:

- Alta: Permet afegir un nou tipus d'article a la nostra col·lecció. Aquest mètode cridarà a un objecte de la base de dades que realitzarà l'alta física en la base de dades.
- Modificar: Permet canviar la descripció d'un tipus d'article existent en la nostra base de dades, mitjançant la crida a un Stored Procedure de la nostra base de dades.
- Baixa: a través del identificador d'aquest tipus d'article, aquest mètode es comunicarà amb un objecte de la base de dades eliminarà físicament el registre. Aquesta acció eliminarà tots els articles que estiguin dins del tipus eliminat.

Classe Fitxa:

Aquesta classe representa a un article dels que treballa l'empresa. Com s'ha comentat anteriorment, tot article estarà relacionat amb un únic tipus d'article per tal de permetre la seva classificació.

Els mètodes d'aquesta classe són:

- Alta: Permet donar d'alta un article en l'aplicació. Aquest mètode cridarà a un objecte de la base de dades que realitzarà l'alta física en la base de dades.
- Modificar: Permet realitzar modificacions sobre un article existent en la nostra base de dades, mitjançant la crida a un Stored Procedure de la nostra base de dades.

- Baixa: a través del identificador d'aquest article, aquest mètode es comunicarà a un objecte de la base de dades que eliminarà físicament el registre.

Classe Proveïdor:

Aquesta classe representa a un proveïdor de tots els que es relacionarà l'empresa i que li proporcionaran articles per a la seva venda. Lògicament, un proveïdor pot aportar més d'un article a l'empresa.

Els mètodes d'aquesta classe són:

- Alta: Permet donar d'alta un proveïdor en l'aplicació. Aquest mètode cridarà a un objecte de la base de dades que realitzarà l'alta física en la base de dades.
- Modificar: Permet realitzar modificacions sobre un proveïdor existent en la nostra base de dades, mitjançant la crida a un Stored Procedure de la nostra base de dades.
- Baixa: a través del identificador d'aquest proveïdor, aquest mètode es comunicarà a un objecte de la base de dades que eliminarà físicament el registre.

Classe LlistatExcel:

Ens permet controlar el llistat que generarà un fitxer Excel a partir de diferents paràmetres.

Com hem dit amb anterioritat, aquest llistat pretén ser el punt de partida per a que l'administrador pugui elaborar un control personalitzat de tot el que controla l'aplicació.

Els mètodes d'aquesta classe són:

Generar: Llença el procés que finalitzarà, en cas d'èxit, amb la generació del llistat.

Classe Grid:

Ens permet controlar les accions que es faran amb les llistes (grids).

Els mètodes d'aquesta classe són:

- Eliminar: Elimina el registre (que correspon a una fila de la llista)
- LlegirTaula: Enllaça la llista amb una taula passada per paràmetre.
- Modificar: Modifica o insereix (segons la consulta passada per paràmetre) un registre (que correspon a una fila de la llista).

Classe Helpers:

Aquesta classe ens proporciona mètodes d'ajuda (help) per a tasques molt repetitives en tota l'aplicació.

Els mètodes d'aquesta classe són:

- FillDropDownList: Permet omplir un combo (llista desplegable) pasta per paràmetre segons una taula també passada per paràmetre. Té diferents constructors.

- GetTable: Retorna el contingut d'una taula segons el nom de la taula passada per paràmetre. També té diversos constructors que permeten fer la mateixa feina amb diferents "matisos" com poden ser, indicar un criteri de selecció, ordenar els registres que retorna ...

Classe Settings:

Aquesta classe s'encarrega de retorna diversos paràmetres de l'aplicació.

Els mètodes d'aquesta classe són:

- SettingsManager: Obté la configuració definida en el fitxer web.config
- ConnectionString: Obté el connectionString definit en el fitxer web.config

Classe ReadImage:

Llegeix la seqüència de bytes d'una imatge emmagatzemada en la taula d'articles segons el identificador del article pasta per paràmetre i l'adapta per la seva visualització en la pàgina d'articles.

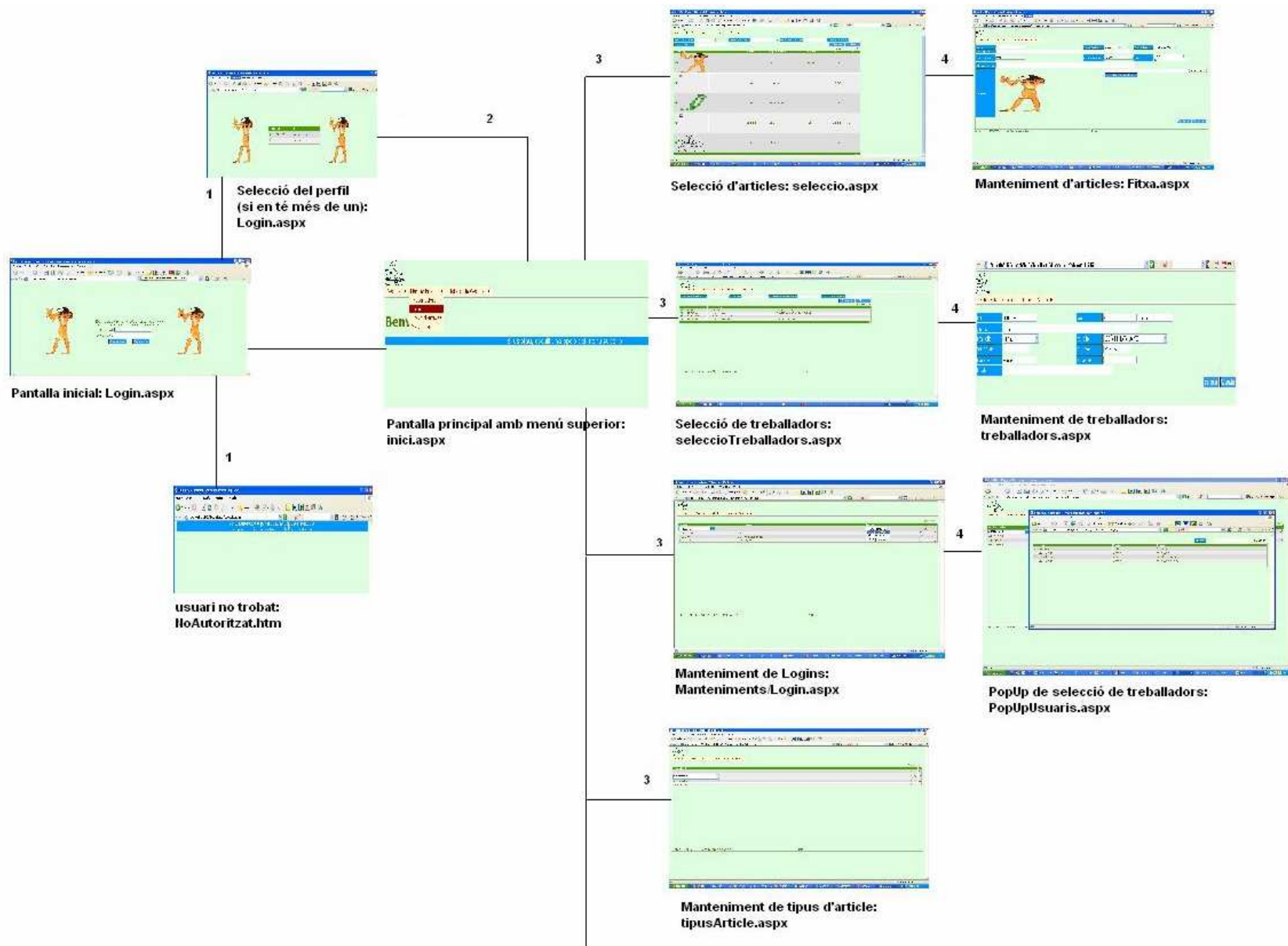
Els mètodes d'aquesta classe són:

- No en té.

6.4.- Disseny de la interfície d'usuari.

Aquesta és una aplicació amb poques pantalles ja que el que cal controlar és bàsicament, com s'ha dit anteriorment, articles, proveïdors, treballadors i alguna taula mestre.

D'aquesta forma, podem definir el fluxe de l'aplicació següent:



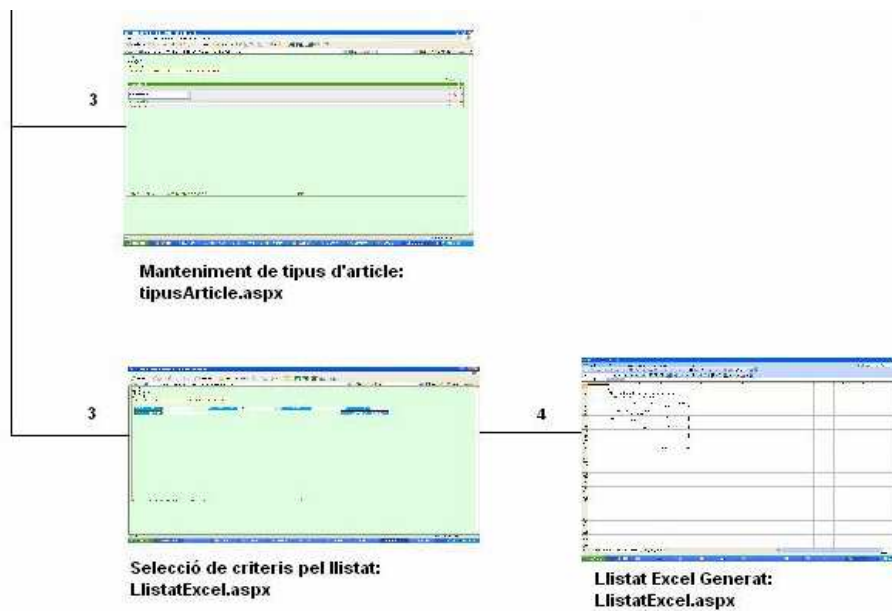


Figura 17. Fluxe de pantalles.

En aquesta seqüència de pantalles partim de la primera pantalla que trobem al executar l'aplicació, que és la de Login, a partir d'aquesta, i sempre que l'usuari indicat estigui donat d'alta en el sistema, podrem accedir a la seqüència de pantalles que s'indica.

L'ordre de la seqüència està indicat amb un número.

Hem de dir que en aquest esquema no hem posat les pantalles corresponent al manteniment de proveïdors per qüestions d'estètica i espai de la imatge ja que aquest manteniment és molt semblant en aparença al de treballadors que sí s'ha incorporat en l'esquema.

De forma una mica més detallada, a continuació anem a mostrar l'aplicació en imatges, seguint la mateixa seqüència indicada en la figura 16.

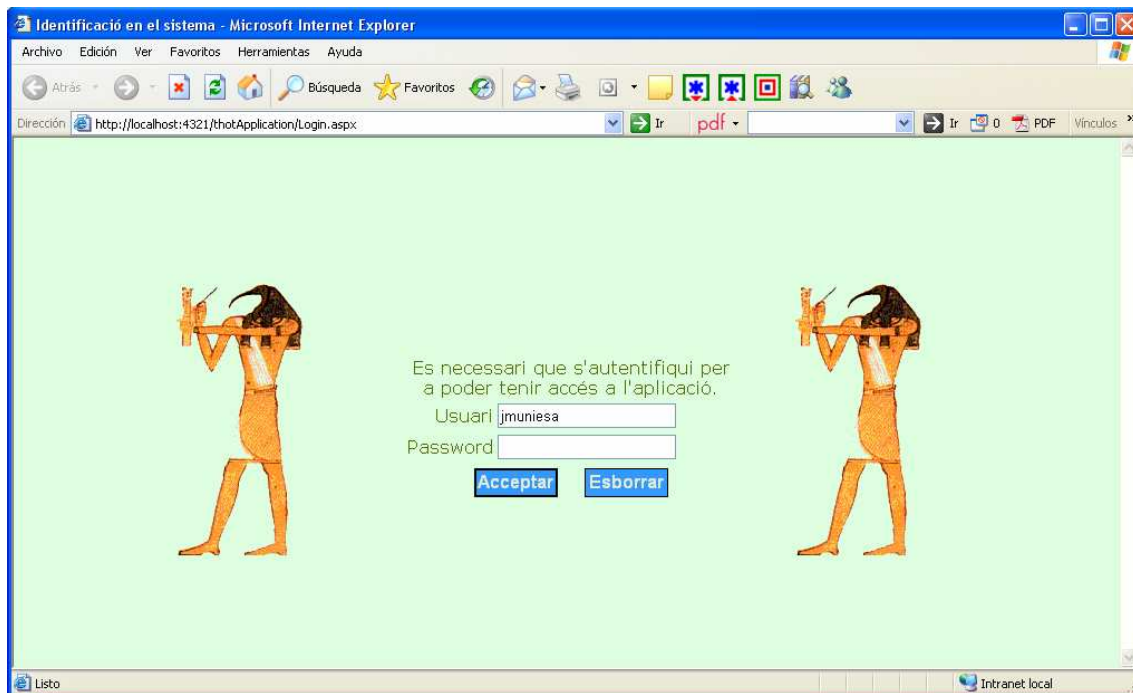


Figura 18. Pantalla de login.

Aquesta pantalla, tal i com s'ha dit abans, és la pantalla inicial i és on l'usuari s'identifica en el sistema.

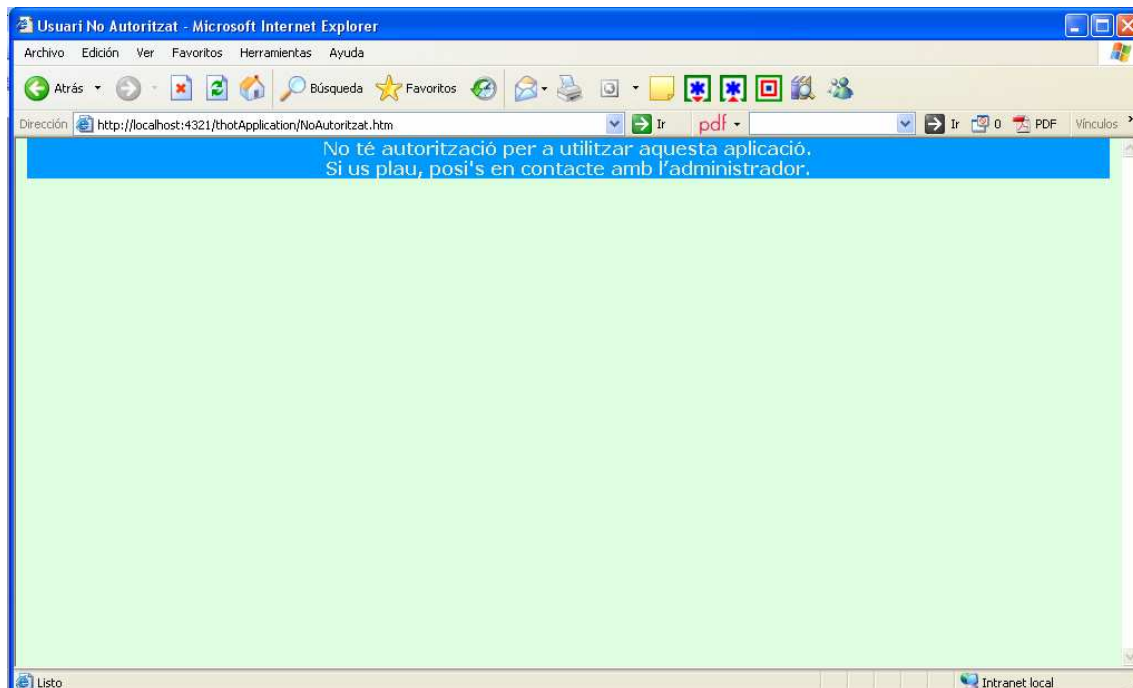


Figura 19. Pantalla de accés no autoritzat.

Si la nostra identificació és errònia anem a parar a aquesta pantalla.

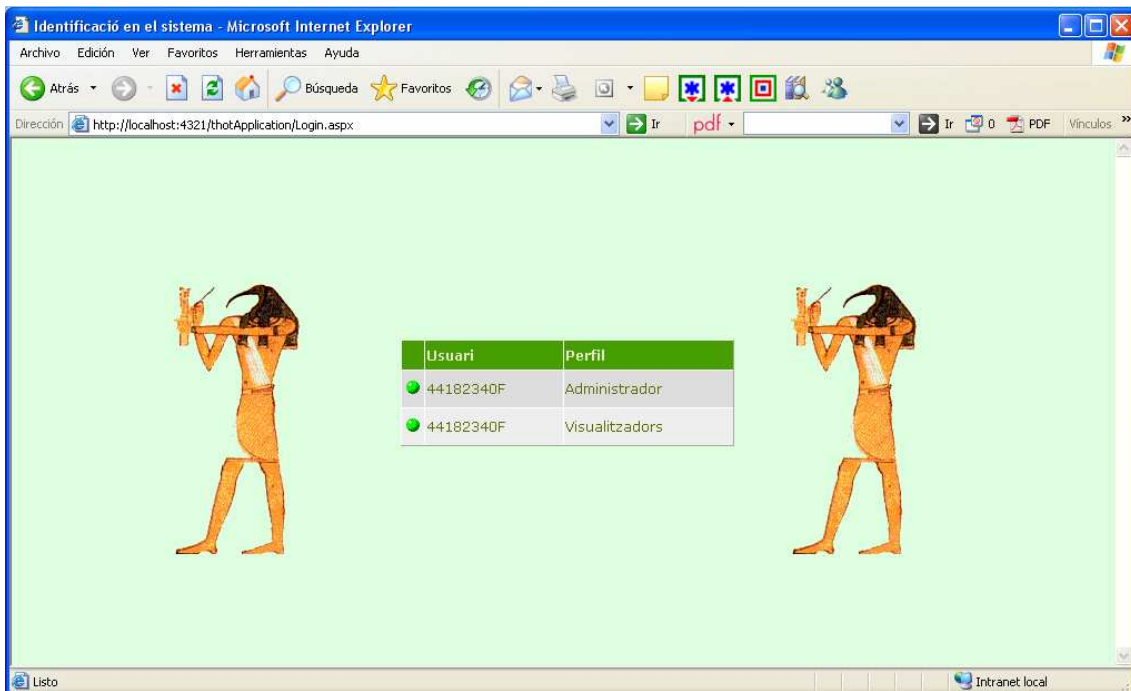


Figura 20. Selecció de perfil.

Si l'usuari introduït en la pantalla de login és correcte arribarem a aquesta pantalla (només en el cas de que l'usuari entrat tingui més d'un perfil, si només té definit un, entrarà directament en la pantalla que es mostra a continuació)



Figura 21. Pantalla d'inici.

Aquesta pantalla és el punt de partida de l'aplicació, a partir d'aquí, caldrà realitzar alguna tria en el menú superior.

Image	Article	Tipus Article	Stock	PVP	IVA
	a	Llibreria	2222	1,0000	1
	aaa	Joguines	1	1,0000	1
	aaaa	Multimèdia	2	1,0000	1

Figura 22. Pantalla de selecció d'articles.

En el cas de que vulguem modificar o eliminar algun article, primer tindrem que cercar-lo a través d'aquesta pantalla de selecció. Des d'aquí també podem donar d'alta nous articles.

Figura 23. Manteniment d'articles.

En aquesta pantalla donarem d'alta, modificarem o eliminarem un article.

Gestió ► Manteniment ► Llistats de Gestió ►

NIF treballador: Nom: Data alta Des de: Data alta Fins:

Total registres: 4

NIF	Nom y cognoms	Població
4412542W	Lluís Moreno	AGRES (03870)
44182339w	victor muniesa	ACEÑA DE LARA, LA (09651)
44182340F	Joan Muniesa Rubio	
44192339W	victor muniesa rubio	AGRA (02409)

Figura 24. Selecció de treballadors.

Aquí podem indicar els criteris de selecció per a cerca un treballador (o uns quants) que s'adaptin al que hem indicat.

Gestió ► Manteniment ► Llistats de Gestió ►

NIF: Nom:

Adreça:

Província: Població:

Telf. Mòbil:

Telf. Fixe:

UserName: Password:

E-mail:

Figura 25. Manteniment de treballadors.

Aquí modificarem, eliminarem o donarem d'alta un treballador.

Gestió ► Manteniment ► Llistats de Gestió ►

Treballador	Descripció	Perfil
44182340F	Joan Muniesa Rubio	Visualitzadors
44182339w	Joan Muniesa Rubio	Operador
44182340F	Joan Muniesa Rubio	Visualitzadors
4412542W	Lluís Moreno	Administrador

Figura 26. Manteniment de logins.

Des d'aquesta única pantalla donarem d'alta, modificarem i eliminarem logins de treballadors (no els treballadors, sin o la relació treballador - perfil)

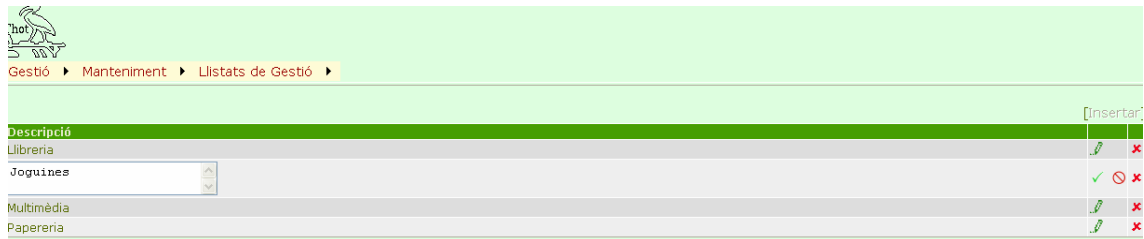


Figura 27. Manteniment de tipus d'article.

Al igual que en el manteniment de logins, podem realitzar totes les operacions des d'aquí.

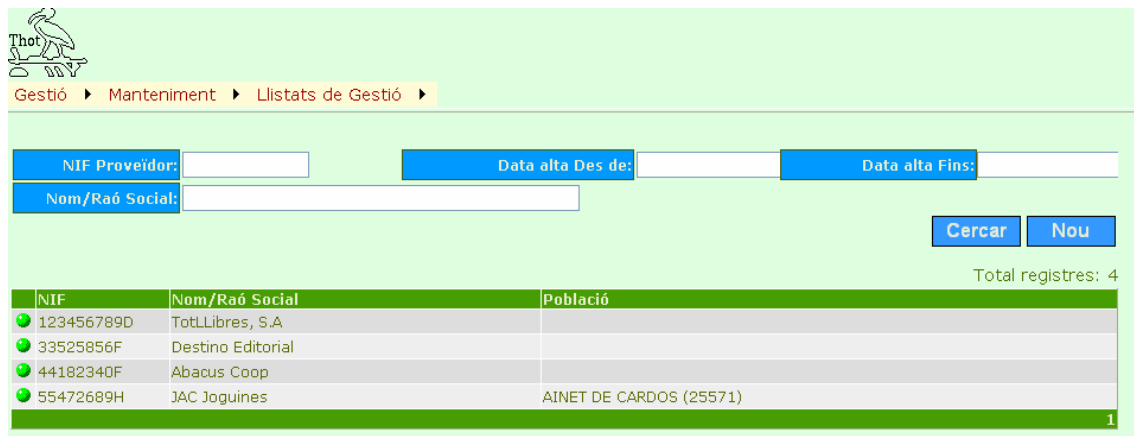


Figura 28. Selecció de proveïdors.

El seu funcionament és igual que la selecció de treballadors. En aquesta pantalla ens podem fixar com, al igual que en la resta de llistes, aquestes estan paginades. Cada pàgina apareix com si fos un link, clicant sobre ell anirem a la pàgina desitjada.

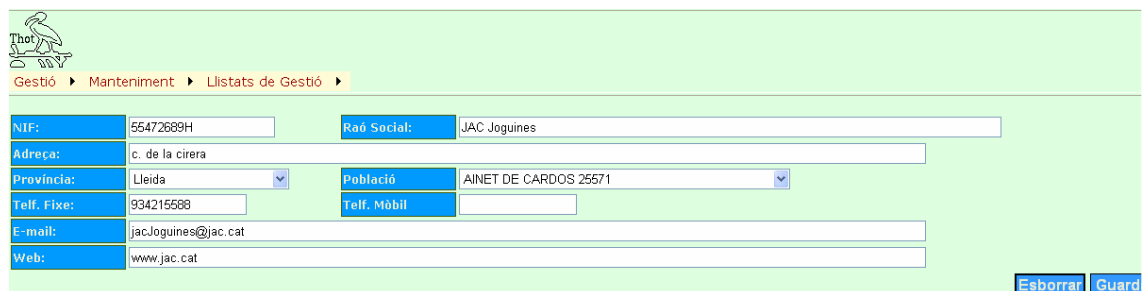


Figura 29. Manteniment de proveïdors.
Idèntic que el manteniment de treballadors

Figura 30. Paràmetres del llistat Excel.

En aquesta pantalla podem indicar els criteris de selecció del llistat Excel.

	A	B	C	D	E	F	G	H
1								
2		Llistat d'articles						
3		Tipus de llistat: llistat d'articles per proveïdor						
4		Data generació llistat: 11/06/2007						
5								
6		Proveedor	Descripció	Stock	Preu Cost	Pvp		
7		Abacus Coop	Los Lunnis	100	12	12		
8		TotLLibres, S	asdsdsd	2	12	1		
9								
10								
11				102	24	13		
12								
13								
14								
15								
16								

Figura 31. Llistat Excel generat

En aquesta imatge podem veure el format del llistat Excel que genera l'aplicació. Podem veure que, encara que és senzill de fer, no dona cap tipus de format final perquè l'usuari que el genera apliqui el que més li agradi o fins i tot algun format que sigui corporatiu.

6.5.- Disseny de la base de dades.

En la següent figura mostrarem el diagrama d'entitat relació (E-R) així com la relació entre entitats.

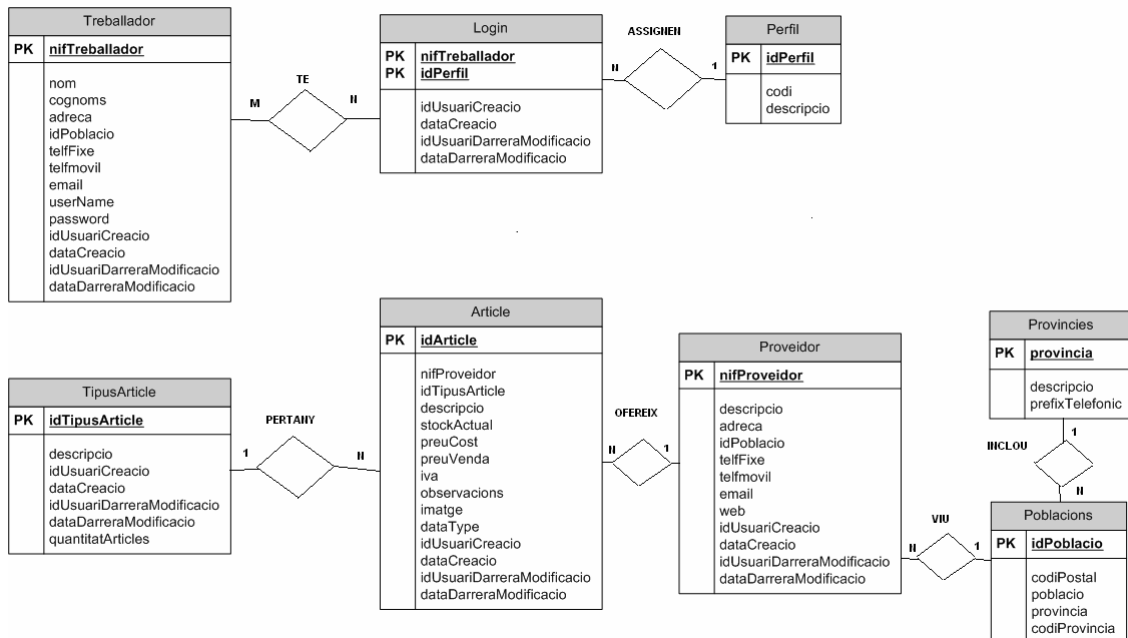


Figura 32. Diagrama E/R

L'accés a qualsevol d'aquestes taules es farà a través dels procediments emmagatzemats (stored procedures) que ens ofereix una base de dades relacional com és el SQL Server de Microsoft degut a que millora el rendiment global de l'aplicació i perquè no serà necessari tornar a compilar i instal·lar l'aplicació al client (sempre i quan tingui els mateixos paràmetres d'entrada i de sortida) ja que simplement re-escriptant el procediment emmagatzemat i executant-lo en la base de dades client aconseguirem realitzar les modificacions.

Hi haurà per a cada taula un procediment emmagatzemat per a la inserció de registres, un altre per la modificació, un altre per l'eliminació, un altre per la consulta i un altre per el llistat dels seus registres.

A continuació descriurem amb cura cada taula i cada camp.

Taula Treballador: Permet emmagatzemar els usuaris de l'empresa que tindran accés a l'aplicació.			
Camp	Clau	Nuls?	Descripció
nifTreballador	PK	No	És la clau primària de la taula. Correspon al NIF del treballador. És guardat correctament ja que en l'aplicació es produirà la seva validació. També correspondrà al login d'entrada a l'aplicació.
Nom	--	No	Nom del treballador
Cognoms	--	No	Cognoms del treballador
Adreca	--	Si	Adreça del treballador
idPoblacio		Si	Població del treballador. Aquesta empresa només treballa en el territori espanyol i no preveu sortir a l'estranger pel que només es controlarà l'existència de codis postals nacionals. Aquest control es durà a terme en l'aplicació.
telfFixe		Si	Telèfon fixe del treballador si en té.
telfMobil		Si	Telèfon mòbil del treballador si en té.
Email		Si	Adreça electrònica del treballador.
userName	--	No	Usuari per accedir a l'aplicació.
Password	--	No	Paraula de pas que li permetrà accedir a l'aplicació. Per decisió expressa de l'empresa, aquest camp no estarà encriptat.
idUsuariCreacio	--	No	Usuari que ha donat d'alta el registre
dataCreacio	--	No	Data i hora en que s'ha donat d'alta el registre
idUsuariDarreraModificacio	--	No	Usuari que ha realitzat la darrera modificació en el registre. Inicialment aquest camp serà igual que el de idUsuariCreacio
dataDarreraModificacio	--	No	Data i hora en la que s'ha realitzat la darrera modificació en el registre. Inicialment aquest camp serà igual que el de dataCreacio

Taula Login: Permet emmagatzemar els diferents perfils dels usuaris de l'empresa que tindran accés a l'aplicació.			
Camp	Clau	Nuls?	Descripció
nifTreballador	PK		Juntament amb el perfil, és la clau primària de la taula. Correspon al NIF del treballador. És guardat correctament ja que en l'aplicació es produirà la seva validació. També correspondrà al login d'entrada a l'aplicació.
idPerfil	PK		Correspon a l'identificador del perfil amb el que l'usuari entrarà a l'aplicació. Un mateix usuari només podrà tenir un perfil de cada tipus.
idUsuariCreacio	--	No	Usuari que ha donat d'alta el registre
dataCreacio	--	No	Data i hora en que s'ha donat d'alta el registre
idUsuariDarreraModificacio	--	No	Usuari que ha realitzat la darrera modificació en el registre. Inicialment aquest camp serà igual que el de idUsuariCreacio
dataDarreraModificacio	--	No	Data i hora en la que s'ha realitzat la darrera modificació en el registre. Inicialment aquest camp serà igual que el de dataCreacio

Taula Perfil: Contindrà els perfils d'accés a l'aplicació. Aquest perfils estaran inicialment definits per l'empresa i seran tres: administrador, visualitzador i gestor. No es preveu que creixi el nombre de registres i per això no tindrà manteniment.

Camp	Clau	Nuls?	Descripció
idPerfil	PK	No	Clau primària del registre. Correspon a un GUID de 32 dígits que identificarà al perfil.
Codi	--	No	Codi de pocs caràcters que permeti identificar el registre. És una codificació interna de l'empresa i no de l'aplicació.
Descripció	--	No	Descripció o nom llarg del perfil

Taula TipusArticles: En aquesta taula guardarem els diferents tipus, classes o famílies d'articles. Cada article ha de estar inclòs dins d'un tipus.

Camp	Clau	Nuls?	Descripció
idTipusArticle	PK	No	Correspon a un GUID de 32 dígits que identificarà el tipus d'article.
Descripcio	--	No	Descripció o nom del tipus d'article.
quantitatArticles	--	No	Aquest camp tindrà un trigger o disparador associat. Cada cop que es doni d'alta, modifiqui o elimini un article s'actualitzarà el comptador associat al tipus d'article al que pertany.
idUsuariCreacio	--	No	Usuari que ha donat d'alta el registre
dataCreacio	--	No	Data i hora en que s'ha donat d'alta el registre
idUsuariDarreraModificacio	--	No	Usuari que ha realitzat la darrera modificació en el registre. Inicialment aquest camp serà igual que el de idUsuariCreacio
dataDarreraModificacio	--	No	Data i hora en la que s'ha realitzat la darrera modificació en el registre. Inicialment aquest camp serà igual que el de dataCreacio
QuantitatArticles	--	No	Quantitat d'articles d'aquest tipus. Inicialment val "0" i, per mitjà d'un trigger, s'anirà incrementant o decremantant

Taula Article: En aquesta taula guardarem els diferents articles que tindrà l'empresa siguin del tipus que siguin.			
Camp	Clau	Nuls?	Descripció
idArticle	PK	No	Correspon a un GUID de 32 dígit que identificarà l'article.
nifProveïdor	FK	No	Identificador del proveïdor que ha proporcionat l'article.
idTipusArticle	FK	No	Indica el tipus d'article que es tracta, la classificació en la que englobem a aquest article.
Descripcio	--	No	Descripció de l'article
stockActual	--	No	Ens indica el stock que disposem de cada article.
preuCost	--	No	Preu que ha pagat l'empresa per aquest article al proveïdor
preuVenda	--	No	Preu de venda de l'article al consumidor
Iva	--	No	Impost que carregarem al article
Observacions	--	Si	Altres dades que vulguem guardar en l'article com per exemple ISBN en el cas de llibres.
Imatge	--	Si	Imatge de l'article
dataType	--	Si	Tipus d'imatge. És necessari per a la seva representació en web.
idUsuariCreacio	--	No	Usuari que ha donat d'alta el registre
dataCreacio	--	No	Data i hora en que s'ha donat d'alta el registre
idUsuariDarreraModificacio	--	No	Usuari que ha realitzat la darrera modificació en el registre. Inicialment aquest camp serà igual que el de idUsuariCreacio
dataDarreraModificacio	--	No	Data i hora en la que s'ha realitzat la darrera modificació en el registre. Inicialment aquest camp serà igual que el de dataCreacio

Taula Proveïdor: Permet emmagatzemar els proveïdors que té l'empresa.			
Camp	Clau	Nuls?	Descripció
nifProveïdor	PK	No	És la clau primària de la taula. Correspon al NIF del proveïdor. És guardat correctament ja que en l'aplicació es produirà la seva validació. També correspondrà al login d'entrada a l'aplicació.
Descripcio	--	No	Raó Social o nom i cognoms del proveïdor.
Adreca	--	No	Adreça del Proveïdor
idPoblacio		No	Població del proveïdor. Aquesta empresa només treballa en el territori espanyol i no preveu sortir a l'estranger pel que només es controlarà l'existència de codis postals nacionals. Aquest control es durà a terme en l'aplicació.
telfFixe		Si	Telèfon fixe del proveïdor.
telfMobil		No	Telèfon mòbil del proveïdor.
Email		Si	Adreça electrònica del treballador.
Web	--	Si	Adreça de la pàgina web del proveïdor. Pot ser útil en el cas que les comandes es puguin fer per web.
idUsuariCreacio	--	No	Usuari que ha donat d'alta el registre
dataCreacio	--	No	Data i hora en que s'ha donat d'alta el registre
idUsuariDarreraModificacio	--	No	Usuari que ha realitzat la darrera modificació en el registre. Inicialment aquest camp serà igual que el de idUsuariCreacio
dataDarreraModificacio	--	No	Data i hora en la que s'ha realitzat la darrera modificació en el registre. Inicialment aquest camp serà igual que el de dataCreacio

Taula Província: Guardarem les províncies de tot el territori espanyol. Lògicament aquesta taula no tindrà manteniment ja que prové l'empresa l'ha comprat a correus. Aquesta empresa té un contracte de manteniment amb correus i en cas d'actualització actualitzaran directament tota la taula mantenint la mateixa codificació. Només serien actualitzacions per corregir noms de províncies o poblacions incorrectes.			
Camp	Clau	Nuls?	Descripció
codiProvíncia	PK	No	Identificarà a la província.
descripcio	--	No	Nom de la província
prefixTelefonic	--	Si	Prefix telefònic

Taula Poblacio: Guardarem les poblacions de tot el territori espanyol. Lògicament aquesta taula no tindrà manteniment ja que prové l'empresa l'ha comprat a correus. La seva actualització segueix els mateixos criteris que en el cas de les províncies.			
Camp	Clau	Nuls?	Descripció
idPoblacio	PK	No	Identificador de la població
codiPostal	--	No	CP de la població
Poblacio	--	No	Descripció de la població.
província	--	No	Codi de la província

Figura 33. Camps de les taules de la base de dades

7.- Conclusions.

Des de el meu punt de vista, crec que en aquest projecte he pogut aplicar moltes de les temàtiques tractades durant tota la Enginyeria en diferents assignatures, des de programació, passant per base de dades a tasques d'anàlisi.

Comparant el llenguatge utilitzat, C#, amb un utilitzat molt en els estudis com és Java, m'ha agradat més treballar amb C#, he vist un entorn que m'ha agradat més i, a de més, al ser de Microsoft i tenir que treballar amb una base de dades també de Microsoft m'ha donat més seguretat.

Crec que m'ha faltat una mica de temps per acabar de donar un disseny més atractiu a l'aplicació però crec que ja és correcte al tractar-se d'un producte d'us intern en un a petita empresa, pel que, al no estar orientada al gran públic crec que és un mal menor.

8.- Línies de desenvolupament futur.

L'objectiu prioritari seria donar més funcionalitat al llistat Excel donant més força a la parametrització.

Seria convenient tenir un fitxer de configuració on es pogués definir el color de les capçaleres, tipus de lletra, imatges corporatives, etc ... per tal d'ajustar més aquest llistat a la filosofia de l'empresa. També es podria pensar en realitzar no un sinó més llistats en format Excel des de diferents punts de vista, des de el producte, des de el proveïdor, etc ...

També es podria pensar en afegir un manteniment d'empreses i realitzar modificacions en les taules per convertir a l'aplicació en un producte multi-empresa.

Un tema que seria d'agrair per l'empresa seria enllaçar d'alguna manera aquesta aplicació amb la seva base de dades central per tal de poder aprofitar les dades que hi ha allí i no tenir que duplicar informació, com per exemple en el cas dels treballadors (i potser també productes, proveïdors ...). Per aconseguir aquesta fita, hi ha diverses alternatives:

- Crear un usuari en la base de dades central i utilitzar-lo en aquesta aplicació per tal de realitzar una connexió
- Programar DTS del SQL Server (Data Transformation Service) per programar tasques nocturnes d'importació de les dades necessàries
- Realitzar un sistema de publicador-suscriptor entre les dues base de dades (aquesta opció seria correcte sobretot si la base de dades central també és SQL Server)

9.- Glossari.

.NET: És un projecte de Microsoft per a crear una nova plataforma de desenvolupament de software amb transparència de xarxes, independència de la plataforma i que permeti un ràpid desenvolupament d'aplicacions.

Ajax: Acrònim d'*Asynchronous JavaScript And XML* (JavaScript i XML asíncrons), és un tècnica de desenvolupament web per a crear aplicacions interactives, Aquestes s'executen en client i mantenen una comunicació asíncrona amb el servidor en un segon pla. D'aquesta forma es possible realitzar canvis sobre la mateixa pàgina sense necessitat de reagregar-la. Això significa un augment d'ela velocitat, interactivitat i usabilitat en la mateixa pàgina.

ATLAS: És el framework de Microsoft que permet utilitzar la tecnologia AJAX.

Callbacks:És semblen a la tecnologia AJAX però només disponible a partir de Framework 2.0

Cas d'us: És una tècnica per a la captura de requeriments potencials d'un nou sistema o una actualització software. Cada cas d'us proporciona un o més escenaris que indiquen com deuria interactuar el sistema amb l'usuari o amb un altre sistema per a aconseguir un objectiu específic.

Diagrama de Gantt: És una popular eina gràfica que te com a objectiu el de mostrar el temps de dedicació previst per a diferents tasques o activitats durant un temps total determinat.

DTS: Aquestes sigles corresponen a "Data Transformation Services" o Servei de Transformació de Dades, és un conjunt d'eines gràfiques i d'objectes programables que permeten als administradors resoldre amb major facilitat problemes de moviment de dades, inclou també la extracció, la transformació i la consolidació de dades de diferents orígens a destins únics o múltiples.

IIS:Aquestes sigles corresponen a Internet Information Service. Son una sèrie de serveis per a ordinadors que funcionen amb Windows. Originalment era part del Option Pack de Windows NT. Després fou integrat en altres sistemes operatius de Microsoft destinats a oferir serveis com Windows 2000 o Windows Server 2003. Windows XP professional inclou una versió limitada de IIS. Els serveis que ofereix són: FTP, SMTP, NNTP i HHTTP/HTTPS

SQL: Structured Query Language (Llenguatge de consulta estructurat). És un llenguatge d'accés a base de dades relacionals que permet especificar diversos tipus d'operacions sobre les mateixes.

SQL Server: És un sistema de gestió de base de dades relacionals (SGBD) basat en el llenguatge SQL capaç de posar en disposició de molts usuaris grans quantitats de dades de manera simultània.

Stored Procedure: Procediment Emmagatzemat. Estan emmagatzemats en la base de dades, es formen amb instruccions SQL i la seva finalitat principal és encapsular l'accés a les base de dades. La seva execució és més ràpida que no les sentències SQL ja que estan precompilats, això significa que abans de la seva execució no s'ha de fer un pre-procés per a conèixer el seu pla d'execució ja que ja es coneix.

10.- Bibliografia.

AJAX The Official Microsoft ASP.NET AJAX Site.

<http://ajax.asp.net/>

SQL Server Express - Instalación y Configuración

http://www.microsoft.com/spanish/msdn/comunidad/mtj.net/voices/MTJ_0005.asp

Curso de AJAX (I) Partimos de los fundamentos - PC Actual

http://www.pc-actual.com/Práctico/Paso_a_paso/Informática_personal/Software/20070117036

Desarrollo de Aplicaciones con Microsoft Ajax 1.0 y Ajax Control Toolkit » SlideShare

<http://www.slideshare.net/pabloesp/desarrollo-de-aplicaciones-con-microsoft-ajax-10-y-ajax-control-toolkit>

SQL Server Usar imágenes de una base de SQL Server 2005

http://www.elguille.info/NET/ADONET/SQL2005Express/imagenes_base_SQLServer.htm

h@nz ...el Geek » Instalar y Configurar ASP .Net AJAX

<http://hancocchi.net/instalar-y-configurar-asp-net-ajax/>

Colabora.NET Usando el CascadingDropDown de AtlasToolkit con una base de datos

http://www.elguille.info/colabora/NET2006/gflores_atlasCDDbd.htm

Using the CascadingDropDown with a Database

<http://ajax.asp.net/ajaxtoolkit/Walkthrough/CCDWithDB.aspx>