2011

TFC – .NET
Memòria

Desenvolupament del Treball de Final de Carrera en .NET
Universitat Oberta de Catalunya

Carles Garcia Bercial
Consultor: Jairo Sarrias Guzman
10/06/2011

Contingut

1.- Resum 3
2.- Introducció 4
2.1.- Justificació i objectius del TFC 4
2.2.- Descripció del projecte 4
2.3.- Alternatives 5
2.4.- Planificació 6
2.4.1.- Fases 6
2.4.2.- Gràfics i taules de la planificació 7
2.4.3.- Seguiment: Planificació inicial i real 8
2.5.- Productes obtinguts 8
2.6.- Tecnologia emprada 8
3.- Anàlisi de requeriments 11
3.1.- Usuaris 11
1.1.- Tipus d'usuaris 11
1.2.- Funcionalitats dels usuaris 11
1.2.1.- Funcionalitats de l'usuari d'aplicació d'escriptori 11
1.2.3.- Funcionalitats de l'administrador 11
4.- Anàlisi del sistema 13
4.1.- Descripció dels actors 13
4.2.- Descripció dels casos d'ús 13
4.2.1.- Cas d'ús comú *Login* al sistema 13
4.2.2.- Casos d'ús de l'Actor 01, Usuari d'aplicació d'escriptori 13
4.2.3.- Casos d'ús de l'Actor 01, Usuari d'aplicació d'escriptori 15
4.3.- Diagrama estàtic de disseny 17
5.- Disseny de l'aplicació 19
5.1.- Arquitectura del sistema: Model de tres capes 19
5.2.- Model de tres capes aplicat a l'aplicació 19
5.2.1.- WPF (Flashcard7, Aplicació) 20
5.2.2.- WCF (Flashcard7.SL, Services Layer) 20
5.2.3.- .NET (Flashcard7.BL, Business Layer) 21
5.2.4.- Linq to SQL (Flashcard7.DAL, Data Access Layer) 22
5.2.5.- SQL Server 2008 (FlashCards7.dbo) 23
5.3.- Base de Dades FlashCards7.dbo 24

5.3.1.- Diagrama de la Base de Dades 24
5.3.2.- Descripció de la Base de Dades 25
6.- Interfície gràfica 27
6.1.- Aplicació d'escriptori 27
6.1.1.- Login 27
6.1.2.- Menú principal 27
6.1.3.- Gestió d'usuaris 28
6.1.4.- Menú de paquets de targetes 29
6.1.5.- Zona d'estudi 30
6.1.6.- Afegir crèdit 30
6.1.7.- Mercat de paquets 31
7.- Objectius aconseguits 32
8.- Treball futur 33
9.- Conclusions 34
10.- Bibliografia 35
10.1.- Llibres 35
10.2.- Recursos online 35
10.3.- Recursos humans 35

1.- Resum

Anàlisi, disseny i implementació d'una aplicació d'escriptori i un servei web per a un **sistema d'estudi amb *flashcards***.

El projecte està dividit en dues parts: L'aplicació d'escriptori i el servei web.
L'aplicació d'escriptori ha de permetre a l'usuari poder utilitzar el sistema per a estudiar amb el format *flashcards* a part de poder comprar paquets de targetes d'estudi, gestionar-los i fer una gestió del seu crèdit.
El servei web ha de fer d'enllaç entre l'aplicació d'escriptori i la base de dades principal proporcionant accés al sistema des de qualsevol lloc amb internet.

Per a desenvolupar el projecte s'ha fet servir la tecnologia **.NET 4.0 de Microsoft**.
L'arquitectura del projecte s'ha fet en 3 capes: Presentació (Aplicació), Negoci (Business) i d'Accés a Dades (Data Access).
El llenguatge de programació principal ha sigut el **C#**.
L'aplicació d'escriptori s'ha desenvolupat en **WPF (*Windows Presentation Foundation*)**.
La comunicació en la capa d'accés a dades s'ha fet mitjançant **WCF (*Windows Communication Foundation*)** i l'accés a dades en sí s'ha fet mitjançant **LINQ to SQL**.
El servei web accedeix a i proporciona dades des d'una base de dades gestionada pel **SGBD SQL Server 2008**.

2.- Introducció

2.1.- Justificació i objectius del TFC

L'objectiu d'aquest TFC és crear una aplicació per a ajudar als estudiants a complementar els seus sistemes d'estudis amb un sistema tan fàcil d'utilitzar com útil, basat en targetes. Amb aquesta aplicació es busca omplir un forat que tenim ara mateix en el nostre mercat que és el de tenir una aplicació basat en el sistema *flashcard* per a ordinadors i, en un futur, per a aparells mòbils.

Amb aquesta aplicació obtindríem que l'usuari es trobés amb un sistema perfecte per a ell i amb l'opció de que cadascú tingués, mitjançant un sistema de comptes online, les seves dades i targetes d'estudi en qualsevol moment en qualsevol lloc.

Tot i això no hem d'oblidar que l'altre objectiu d'aquest TFC és l'aprenentatge de les tecnologies Microsoft. Utilitzant el IDE (*Integrated Development Environment*, Entorn de Desenvolupament Integrat) Visual Studio 2010 he profunditzat els coneixements que ja tenia en el llenguatge C# i sobretot he après en el àmbit de WPF, WCF i l'interessantíssim món que m'ha sembla el Linq to SQL, tot això basat en el marc de treball .NET 4.0.

2.2.- Descripció del projecte

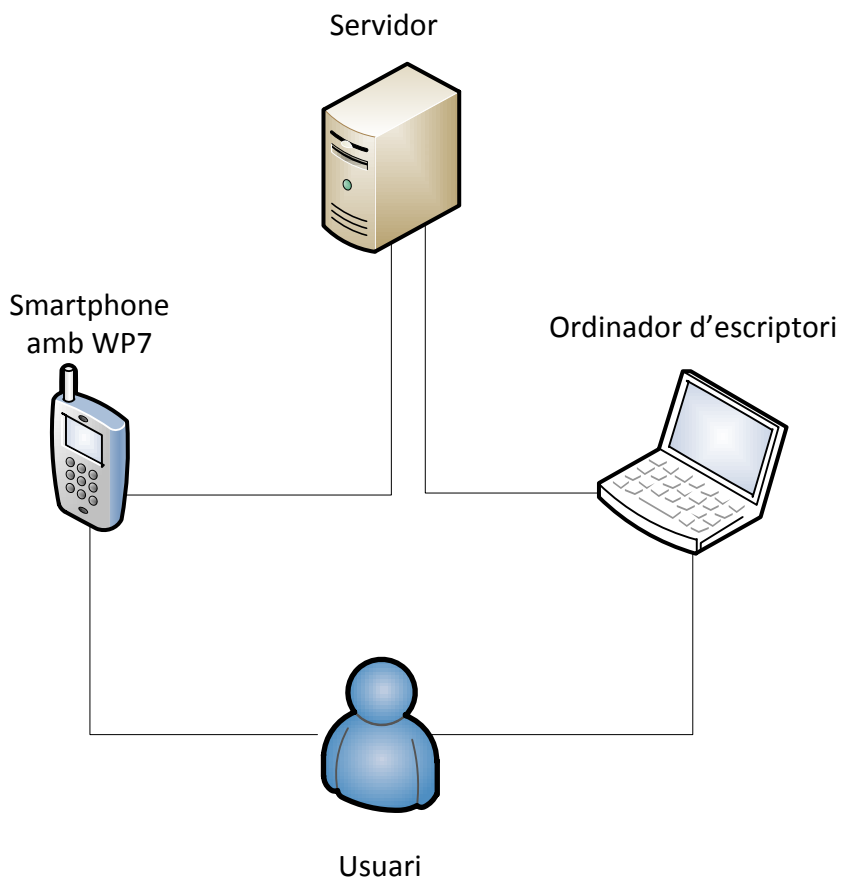
El projecte consisteix en la creació d'unes aplicacions per l'ajuda de l'estudi mitjançant el sistema de *flashcards* o targetes d'estudi i està definit en tres parts: una aplicació d'escriptori, una aplicació mòbil i un servei web.

Totes les parts del projecte estaran desenvolupades en .NET però cadascuna utilitzant diferents tecnologies dins d'aquest marc.

Per una part, tindrem el servei web que estarà desenvolupat amb la tecnologia WCF (*Windows Communication Foundation*). Els usuaris de les aplicacions de mòbil i escriptori atacaran aquest servei per a fer servir els seus recursos.

L'aplicació per a mòbils estarà dirigida per a dispositius equipats amb el sistema operatiu WP7 (*Windows Phone 7*) i estarà desenvolupat principalment utilitzant *Silverlight*. Des d'aquesta aplicació l'usuari podrà accedir a la seva compta i a les seves targetes per a poder estudiar. Encara que l'aplicació tindrà opcions online la majoria de les funcions seran offline per a no acaparar la xarxa del dispositiu.

I com a últim tenim l'aplicació d'escriptori que estarà desenvolupada utilitzant WPF (*Windows Presentation Foundation*). Aquí és on l'usuari es podrà connectar a la seva compte, comprar paquets de targetes d'estudi, posar saldo al seu compte i, per descomptat, estudiar amb les targetes.



Les tres parts del projecte encara que són diferents aplicacions funcionen i es coordinen les unes amb les altres per tal de funcionar com una de sola.

Amb aquest conjunt d'aplicacions qualsevol persona pot estudiar estigui on estigui i de la forma més còmoda gràcies a un sistema per targetes que ara serà molt més còmode d'utilitzar.

2.3.- Alternatives

En aquest moment en el mercat hi ha algunes alternatives però cap ofereix la mateixa metodologia d'estudi i encara menys la possibilitat de poder tenir el teu propi sistema d'estudi estiguis on estiguis.

Un exemple podria ser http://www.braineos.com/ però com es pot veure encara està en fase de desenvolupament i encara que té diferents "jocs" per estudiar, no té gaire material educatiu.

Per suposat tenim el sistema de targetes físiques però clarament cap alternativa digital comparable amb aquesta.

Aquest projecte seria un avenç i una novetat en el conjunt d'eines digitals per a l'educació i l'estudi.

2.4.- Planificació

2.4.1.- Fases

El projecte consta de 4 fases definides per unes fites a realitzar.

2.4.1.1.- Fase 1: Pla de treball i anàlisi de requeriments

03/03/2011 – 16/03/2011

Després de decidir el projecte es passa a la realització del document [Pla de treball] on es fa una breu descripció del projecte i es defineixen els objectius generals d'aquest. Es fa una especificació dels requeriments del projecte així com les fases en que es divideix la feina. En aquesta fase també es procedeix a la descàrrega del software per a desenvolupar el projecte i la posta a punt d'aquests.

2.4.1.2.- Fase 2: Especificació i disseny

17/03/2011 – 11/04/2011

Durant tota aquesta fase es realitza un estudi de les tecnologies a utilitzar i un aprenentatge d'aquestes. Es realitza la especificació i el disseny de la BBDD, el servei web, l'aplicació per escriptori i l'aplicació mòbil.

2.4.1.3.- Fase 3: Implementació

12/04/2011 – ~~23/05/2011~~ 2/06/2011

En aquesta fase es dedica una bona part del temps en la implementació dels dissenys creats del projecte en les diferents tecnologies que engloben el .NET. Un cop feta la implementació s'utilitza gairebé el mateix temps en fer un testeig de les aplicacions per deixar-les lliures d'errors i s'intenta millorar algunes parts. També es fa un manual d'instal·lació que es lliura juntament amb el codi.

Aquesta fase s'allarga una mica més per uns problemes greus que sorgeixen en la implementació de l'aplicació mòbil fent-ne, finalment, impossible el desenvolupament total d'aquesta.

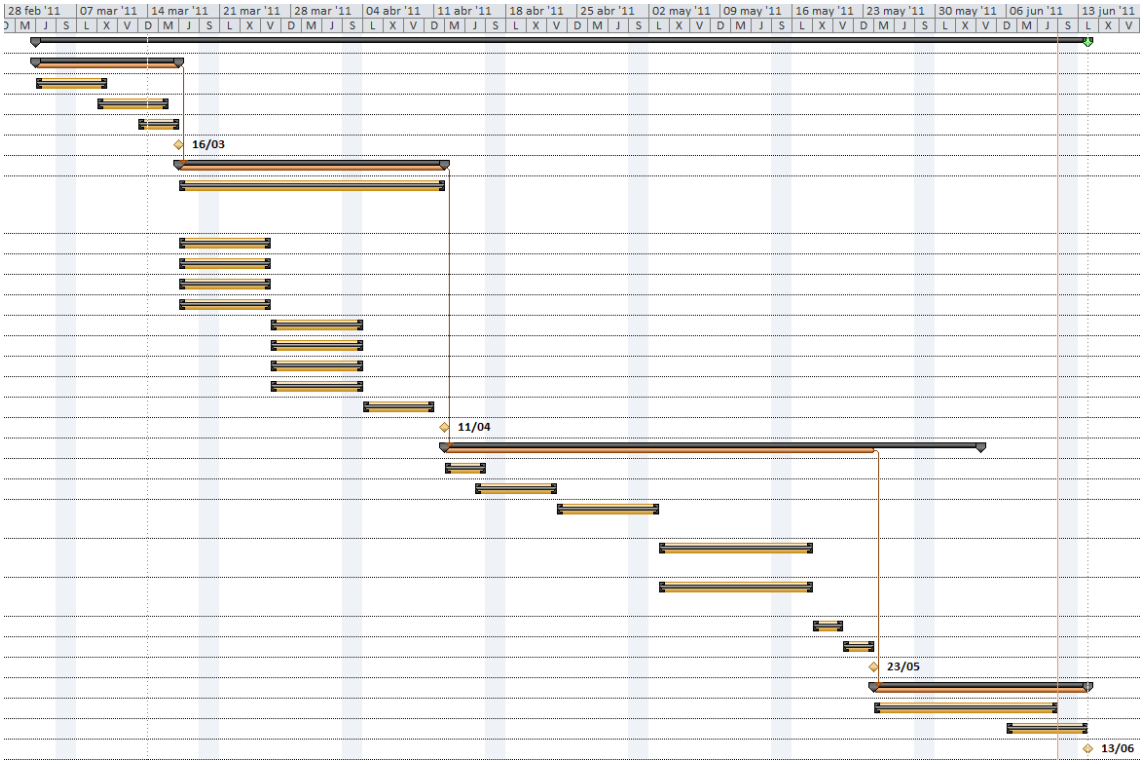
2.4.1.4.- Fase 4: Memòria i presentació virtual

24/05/2011 – 13/06/2011

Fase final on s'acaba i es munta la Memòria del projecte on es veu reflectit tot el treball realitzat del projecte així com una definició precisa. També es crea una presentació virtual per al jurat explicant què i com és el projecte mitjançant un video.

2.4.2.- Gràfics i taules de la planificació

		Mod de	Nombre de tarea	Duración	Comienzo	Fin
1			**TFC . NET**	**73 días**	**jue 03/03/11**	**lun 13/06/11**
2	✓		**PAC 1 - Pla de treball**	**10 días**	**jue 03/03/11**	**mié 16/03/11**
3	✓		Proposta d'enunciat	5 días	jue 03/03/11	mié 09/03/11
4	✓		Desenvolupament Pla de treball	5 días	mié 09/03/11	mar 15/03/11
5	✓		Descàrrega de software	4 días	dom 13/03/11	mié 16/03/11
6	✓		Entrega Pla de treball	1 día	mié 16/03/11	mié 16/03/11
7			**PAC 2 - Especificació i disseny**	**18 días**	**jue 17/03/11**	**lun 11/04/11**
8	✓		Adquirir coneixements de les diferents tecnologies (WCF, WPF, Silverlight, LINQ)	18 días	jue 17/03/11	lun 11/04/11
9	✓		Especificació BBDD	7 días	jue 17/03/11	vie 25/03/11
10	✓		Especificació Web Service	7 días	jue 17/03/11	vie 25/03/11
11	✓		Especificació Aplicació Escriptori	7 días	jue 17/03/11	vie 25/03/11
12	✓		Especificació Aplicació Mòbil	7 días	jue 17/03/11	vie 25/03/11
13	✓		Disseny BBDD	7 días	sáb 26/03/11	dom 03/04/11
14	✓		Disseny Web Service	7 días	sáb 26/03/11	dom 03/04/11
15	✓		Disseny Aplicació Escriptori	7 días	sáb 26/03/11	dom 03/04/11
16	✓		Disseny Aplicació Mòbil	7 días	sáb 26/03/11	dom 03/04/11
17	✓		Elaboració PAC 2	6 días	lun 04/04/11	dom 10/04/11
18			Entrega PAC 2	1 día	lun 11/04/11	lun 11/04/11
19	✓		**PAC 3 - Implementació**	**38,5 días**	**mar 12/04/11**	**vie 03/06/11**
20	✓		Implementació BBDD	4 días	mar 12/04/11	vie 15/04/11
21	✓		Implementació Web Service	6 días	vie 15/04/11	vie 22/04/11
22	✓		Implementació Aplicacions d'Escriptori i Mòbil	7 días	sáb 23/04/11	lun 02/05/11
23	✓		Testeig i probes de la implementació	11 días	mar 03/05/11	mar 17/05/11
24	✓		Correcció de bugs i millores de la implementació	11 días	mar 03/05/11	mar 17/05/11
25	✓		Desplegament	3 días	mié 18/05/11	vie 20/05/11
26	✓		Elaboració Manual d'Instal·lació	2 días	sáb 21/05/11	lun 23/05/11
27	✓		Entrega PAC 3	1 día	lun 23/05/11	lun 23/05/11
28	✓		**PAC 4 - Memòria i presentació virtual**	**15 días**	**mar 24/05/11**	**lun 13/06/11**
29	✓		Elaboració de la memòria	14 días	mar 24/05/11	vie 10/06/11
30	✓		Elaboració Presentació Virtual	6 días	lun 06/06/11	lun 13/06/11
31	✓		Entrega PAC 4	1 día	lun 13/06/11	lun 13/06/11



2.4.3.- Seguiment: Planificació inicial i real

Les diferències entre la planificació que es va fer proposar al principi i la que finalment s'ha seguit ha estat bàsicament la mateixa i, sense comptar imprevistos que en tots els projectes succeeixen, totes les fites s'han aconseguit i entregat a temps menys la tercera que, per problemes tècnics greus no es va poder fer al dia i es va haver de fer una setmana més tard.

2.5.- Productes obtinguts

Durant la realització del TFC s'han anat obtenint els documents detallats a continuació:

- **Document Pla de treball** (PAC 1): És un document on es recullen els objectius, els requeriments i la planificació del TFC.
- **Document Anàlisi i disseny** (PAC 2): És un document on es fa una proposta formal dels requeriments, els casos d'ús dels actors, el model conceptual de les classes, l'arquitectura del sistema, el disseny de la base de dades i els prototips dels dissenys de la interfície gràfica.
- **Implementació** (PAC3): És un paquet de diferents documents i està compost per el codi font del projecte (una solució de Visual Studio 2010 amb diferents projectes a dins), un fitxer de recuperació de base de dades (en format .bak) per al SQL Server 2008, un manual de l'usuari, un manual de l'administrador, un instal·lador del programari i un document amb comentaris i instruccions sobre la implementació.
- **Memòria final i presentació** (PAC 4): Un paquet amb el document de memòria del projecte i una presentació en format vídeo per a complementar la memòria.

2.6.- Tecnologia emprada

Per al desenvolupament del projecte s'han utilitzat diferents tecnologies de Microsoft comentades breument a continuació.

.NET 4.0 és un marc de treball que inclou una gran llibreria i recolza diferents llenguatges de programació que permeten interoperabilitat entre ells. La llibreria .NET està disponible a tots els llenguatges de programació que el .NET suporta (en el nostre cas **C#**).

La llibreria de classes base del *framework* proveeix interfícies d'usuari, accés a dades, connectivitat amb bases de dades, criptografia, desenvolupament d'aplicacions web, algoritmes numèrics, i comunicació en xarxa. El programador és l'encarregat de barrejar aquestes llibreries amb el seu propi codi per crear les aplicacions.

L'entorn de treball **Visual Studio 2010** és l'eina ideal per a treballar amb aquest marc de treball gràcies a totes les ajudes que proporciona un entorn de treball dedicat com aquest.

El **Linq** (*Language Integrated Query*) és un component del .NET que afegeix capacitats natives de consulta de dades als llenguatges de .NET. Ens ha permès tractar amb les dades relacionals provinents d'una base de dades de SQL Server com si fossin objectes (classes) de .NET.

El **SQL Server 2008** és un SGBD (Sistema Gestor de Bases de Dades relacionals) basat en el llenguatge SQL. És una eina capaç de proporcionar grans quantitats de dades a molts usuaris simultàniament.

Per a administrar el SQL Server 2008 utilitzarem una eina integrada que ens ofereix un entorn gràfic i un conjunt d'eines força simples i potents per a crear i modificar convenientment la base de dades anomenat SQL Server Management Studio.

El **WPF** (*Windows Presentation Foundation)* és un subconjunt del marc de treball o *framework* .NET 4.0 (encara que hi és present des de la versió 3.0) i si s'han fet servir tecnologies com WinForms o ASP.NET ràpidament es poden veure les similituds. És un entorn que treballa directament amb DirectX i arrelat molt profundament a la presentació gràfica de les aplicacions ja que treballa amb un motor basat en càlculs vectorials.

El WPF es composa de dues parts, el llenguatge declaratiu de marcat o *markup*, **XAML** (*Extensible Application Markup Language*), i el codi subjacent o *code-behind*.

Per al desenvolupament d'aquesta part de l'aplicació s'ha utilitzat el Microsoft **Expression Blend 4** per la part de XAML ja que aquesta eina està enfocada precisament en l'apartat de disseny de la interfície gràfica i per al *code-behind* s'ha combinat el mateix Expression Blend i el Visual Studio 2010.

El **WCF** (*Windows Communication Foundation*) és un kit de desenvolupament de programari per desenvolupar i desplegar serveis en sistemes Windows. Proveeix un entorn en temps d'execució per a serveis que permet exposar tipus CLR (*Common Language Runtime*) com a serveis i utilitzar altres serveis com a tipus CLR. WCF és una implementació de Microsoft per a un conjunt d'estàndards de la indústria definint interaccions entre serveis, conversions de tipus, càlcul de referències i la gestió de varis protocols.

Aquest kit és part del *framework* .NET 4.0 encarà que hi ha estat present des del .NET 3.0. L'actual és la tercera revisió del WCF i des de l'original de l'any 2006 s'hi ha afegit eines per a la

gestió de instàncies de serveis, crides asíncrones, gestió de transaccions, més seguretat i d'altres.

3.- Anàlisi de requeriments

3.1.- Usuaris

Definició dels usuaris dins el projecte. Es farà una descripció dels usuaris i la interacció d'aquests amb les diferents aplicacions.

1.1.- Tipus d'usuaris

Hi haurà dos tipus d'usuari segons l'aplicació que utilitzin i un tipus d'usuari especial:

- Usuari d'aplicació d'escriptori: Aquests seran capaços d'accedir a la seva compta, afegir saldo per a poder fer compres online, navegar entre els paquets de targetes disponibles en el mercat i comprar-ne de nous i utilitzar el sistema d'estudi.
- Administrador: Aquest usuari serà l'únic que mitjançant un apartat especial de l'aplicació d'escriptori accedirà a la gestió d'usuari. Des d'aquest apartat podrà crear, modificar i eliminar usuaris.

1.2.- Funcionalitats dels usuaris

Cada usuari del sistema té les seves funcionalitats ben definides encara que algunes d'aquestes funcionalitats són compartides per diferents usuaris. Les funcionalitats compartides, encara que són les mateixes accions poden donar accés a d'altres funcionalitats.

Passarem a fer la descripció de les funcionalitats segons l'usuari.

1.2.1.- Funcionalitats de l'usuari d'aplicació d'escriptori

***Login* al sistema**: L'usuari es podrà identificar al sistema per a poder començar a utilitzar l'aplicació.

Consulta de paquets de targetes: Es podrà navegar entre els paquets que l'usuari disposi de forma online.

Compra de paquets de targetes: En l'apartat de mercat de l'aplicació, l'usuari podrà comprar nous paquets.

Gestionar el crèdit: Mitjançant codis que es podran comprar per internet o en tendes es podrà augmentar el saldo de la compta.

Estudi: Aquesta opció permet a l'usuari fer ús de les targetes per a estudiar.

1.2.3.- Funcionalitats de l'administrador

***Login* al sistema**: L'usuari es podrà identificar al sistema per a poder començar a utilitzar l'aplicació.

Consulta de paquets de targetes: Es podrà navegar entre els paquets que l'usuari disposi de forma online.

Compra de paquets de targetes: En l'apartat de mercat de l'aplicació, l'usuari podrà comprar nous paquets.

Gestionar el crèdit: Mitjançant codis que es podran comprar per internet o en tendes es podrà augmentar el saldo de la compta.

Estudi: Aquesta opció permet a l'usuari fer ús de les targetes per a estudiar.

Gestió d'usuaris: Podrà supervisar les dades dels usuaris i podrà accedir a la informació d'aquests per a solucionar conflictes. Podrà crear, modificar o eliminar usuaris.

4.- Anàlisi del sistema

Es farà una descripció dels casos d'ús més importants de les aplicacions. Es mostrarà un esquema i la seva definició.

4.1.- Descripció dels actors

A continuació hi ha una descripció dels actors a tenir en compte.

Actor 01	Usuari d'aplicació d'escriptori
Descripció	Aquest actor serà el que accedirà al sistema mitjançant l'aplicació d'escriptori.
Comentaris	Cap.

Actor 02	Administrador
Descripció	Accedirà al sistema des de l'aplicació d'escriptori.
Comentaris	Té privilegis totals sobre el sistema.

En moltes ocasions coincidirà que una mateixa persona estigui classificada en el paper d'Actor 01 i d'Actor 02.

4.2.- Descripció dels casos d'ús

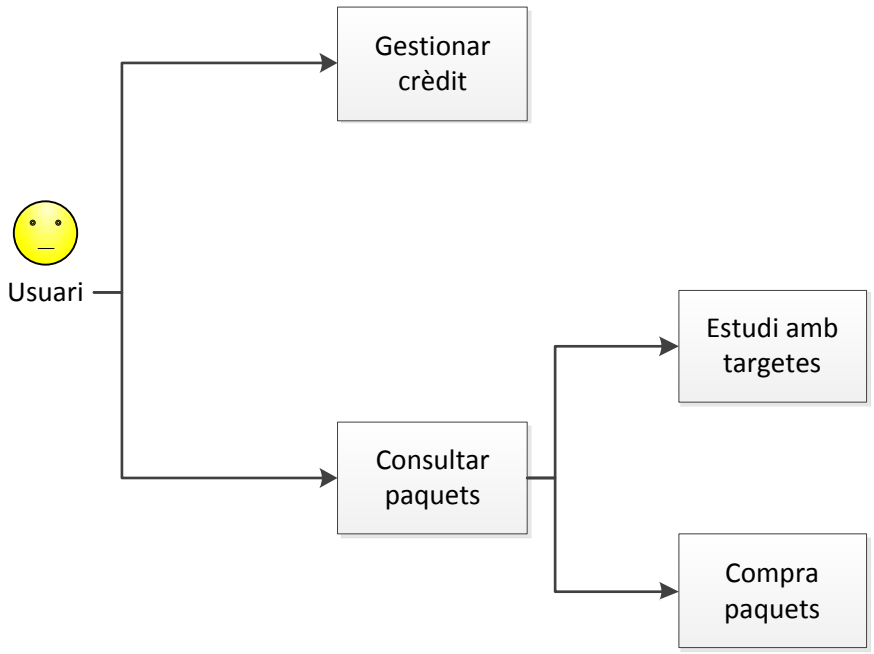
4.2.1.- Cas d'ús comú *Login* al sistema

Tots els usuaris hauran de validar-se contra el sistema per a poder accedir als seus comptes i poder accedir a algunes o totes les opcions de les aplicacions.

- **Resum**: Un usuari s'identifica davant el sistema per a accedir a aquest.
- **Actors**: Usuaris d'aplicació d'escriptori i administradors.
- **Precondició**: L'actor ha d'existir per al sistema.
- **Postcondició**: L'actor aconsegueix credencials en el sistema.
- **Descripció**: L'actor es validarà al sistema i tindrà accés a les funcionalitats d'aquest depenent del nivell de privilegis que aquest tingui.

4.2.2.- Casos d'ús de l'Actor 01, Usuari d'aplicació d'escriptori

Descripció dels casos d'ús on l'actor és un usuari d'aplicació d'escriptori.



4.2.2.1.- Cas d'ús Gestionar crèdit

- **Resum**: Un usuari valida un codi per a modificar el seu crèdit.
- **Actors**: Usuari d'aplicació d'escriptori, Administrador.
- **Precondició**: L'usuari ha d'estar validat al sistema. El codi no ha estat utilitzat.
- **Postcondició**: El crèdit de l'usuari es veu modificat.
- **Descripció**: L'usuari introduirà un codi a l'apartat de administració de crèdit. Aquest codi es validarà al sistema i si no ha estat utilitzat amb anterior i és un codi vàlid el valor d'aquest codi es sumarà al crèdit de l'usuari

4.2.2.2.- Cas d'ús Comprar paquets

- **Resum**: Un usuari, mitjançant el mercat de l'aplicació, compra paquets nous de targetes.
- **Actors**: Usuari d'aplicació d'escriptori, Administrador.
- **Precondició**: El paquet de targetes no ha d'haver estat comprat amb anterioritat.
- **Postcondició**: L'usuari disposa d'un nou paquet de targetes.
- **Descripció**: L'usuari comprarà un nou paquet de targetes en el mercat i aquest, si no ha estat comprat amb anterioritat, quedarà enregistrat en la BBDD. Aquesta acció quedarà enregistrada en la BBDD i a partir d'aquí l'usuari podrà utilitzar aquest recurs.

4.2.2.3.- Cas d'ús Consultar paquets

- **Resum**: L'usuari accedeix a tots els paquets de targetes que aquest hagi comprat.
- **Actors**: Usuari d'aplicació d'escriptori, Administrador.
- **Precondició**: Existeix algun paquet de targetes enregistrat per a l'usuari.
- **Postcondició**: Cap.
- **Descripció**: Podrà navegar per un menú on podrà accedir a la informació de tots els paquets de targetes que té.

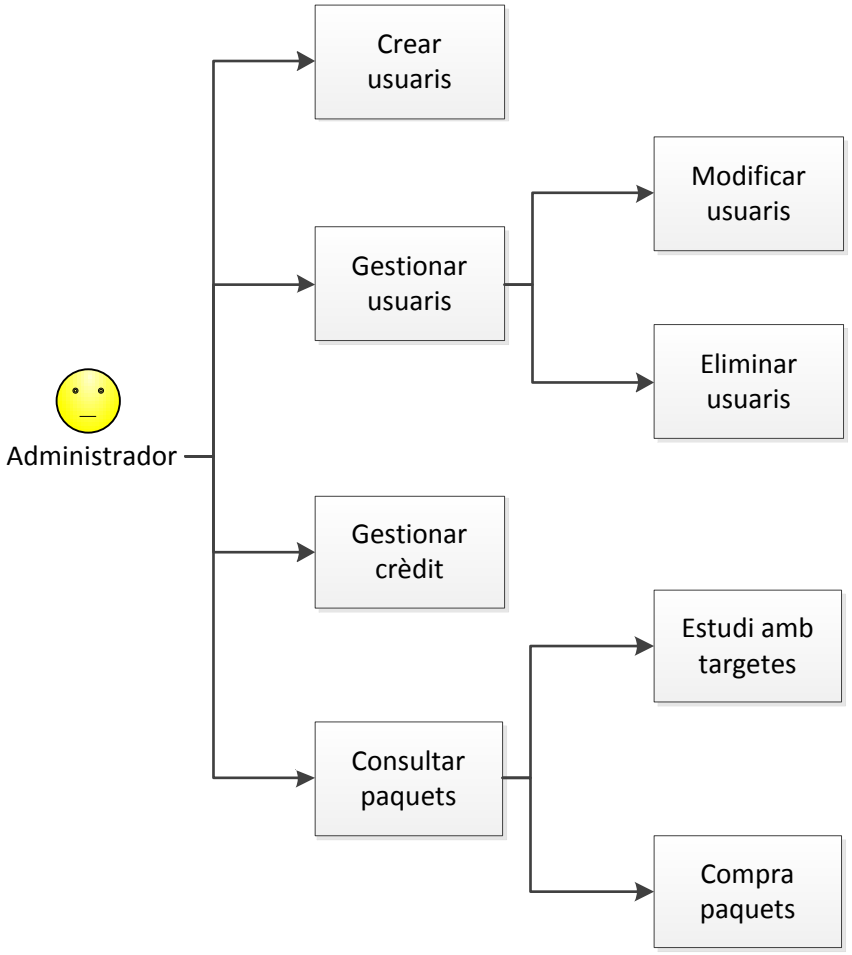
4.2.2.4.- Cas d'ús Estudi amb targetes

- **Resum**: S'accedeix a un paquet de targetes per a fer-ne ús per a l'estudi en xarxa.

- **Actors**: Usuari d'aplicació d'escriptori, Administrador.
- **Precondició**: Existeix algun paquet de targetes registrat a la xarxa.
- **Postcondició**: Cap.
- **Descripció**: L'usuari podrà fer ús dels paquets de targetes que estiguin disponibles de forma online o en xarxa per a estudiar.

4.2.3.- Casos d'ús de l'Actor 01, Usuari d'aplicació d'escriptori

Descripció dels casos d'ús on l'actor és un usuari d'aplicació d'escriptori.



4.2.3.1.- Cas d'ús Gestionar crèdit

- **Resum**: Un usuari valida un codi per a modificar el seu crèdit.
- **Actors**: Usuari d'aplicació d'escriptori, Administrador.
- **Precondició**: L'usuari ha d'estar validat al sistema. El codi no ha estat utilitzat.
- **Postcondició**: El crèdit de l'usuari es veu modificat.
- **Descripció**: L'usuari introduirà un codi a l'apartat de administració de crèdit. Aquest codi es validarà al sistema i si no ha estat utilitzat amb anterior i és un codi vàlid el valor d'aquest codi es sumarà al crèdit de l'usuari

4.2.3.2.- Cas d'ús Comprar paquets

- **Resum**: Un usuari, mitjançant el mercat de l'aplicació, compra paquets nous de targetes.
- **Actors**: Usuari d'aplicació d'escriptori, Administrador.

- **Precondició**: El paquet de targetes no ha d'haver estat comprat amb anterioritat.
- **Postcondició**: L'usuari disposa d'un nou paquet de targetes.
- **Descripció**: L'usuari comprarà un nou paquet de targetes en el mercat i aquest, si no ha estat comprat amb anterioritat, quedarà enregistrat en la BBDD. Aquesta acció quedarà enregistrada en la BBDD i a partir d'aquí l'usuari podrà utilitzar aquest recurs.

4.2.3.3.- Cas d'ús Consultar paquets

- **Resum**: L'usuari accedeix a tots els paquets de targetes que aquest hagi comprat.
- **Actors**: Usuari d'aplicació d'escriptori, Administrador.
- **Precondició**: Existeix algun paquet de targetes enregistrat per a l'usuari.
- **Postcondició**: Cap.
- **Descripció**: Podrà navegar per un menú on podrà accedir a la informació de tots els paquets de targetes que té.

4.2.3.4.- Cas d'ús Estudi amb targetes

- **Resum**: S'accedeix a un paquet de targetes per a fer-ne ús per a l'estudi en xarxa.
- **Actors**: Usuari d'aplicació d'escriptori, Administrador.
- **Precondició**: Existeix algun paquet de targetes registrat a la xarxa.
- **Postcondició**: Cap.
- **Descripció**: L'usuari podrà fer ús dels paquets de targetes que estiguin disponibles de forma online o en xarxa per a estudiar.

4.2.3.5.- Cas d'ús Crear usuari

- **Resum**: Es crea un usuari nou al sistema.
- **Actors**: Administrador.
- **Precondició**: No existeix aquest usuari al sistema.
- **Postcondició**: Hi ha un usuari nou.
- **Descripció**: Es crearà un usuari nou que serà introduït a la BBDD.

4.2.3.6.- Cas d'ús Modificar usuaris

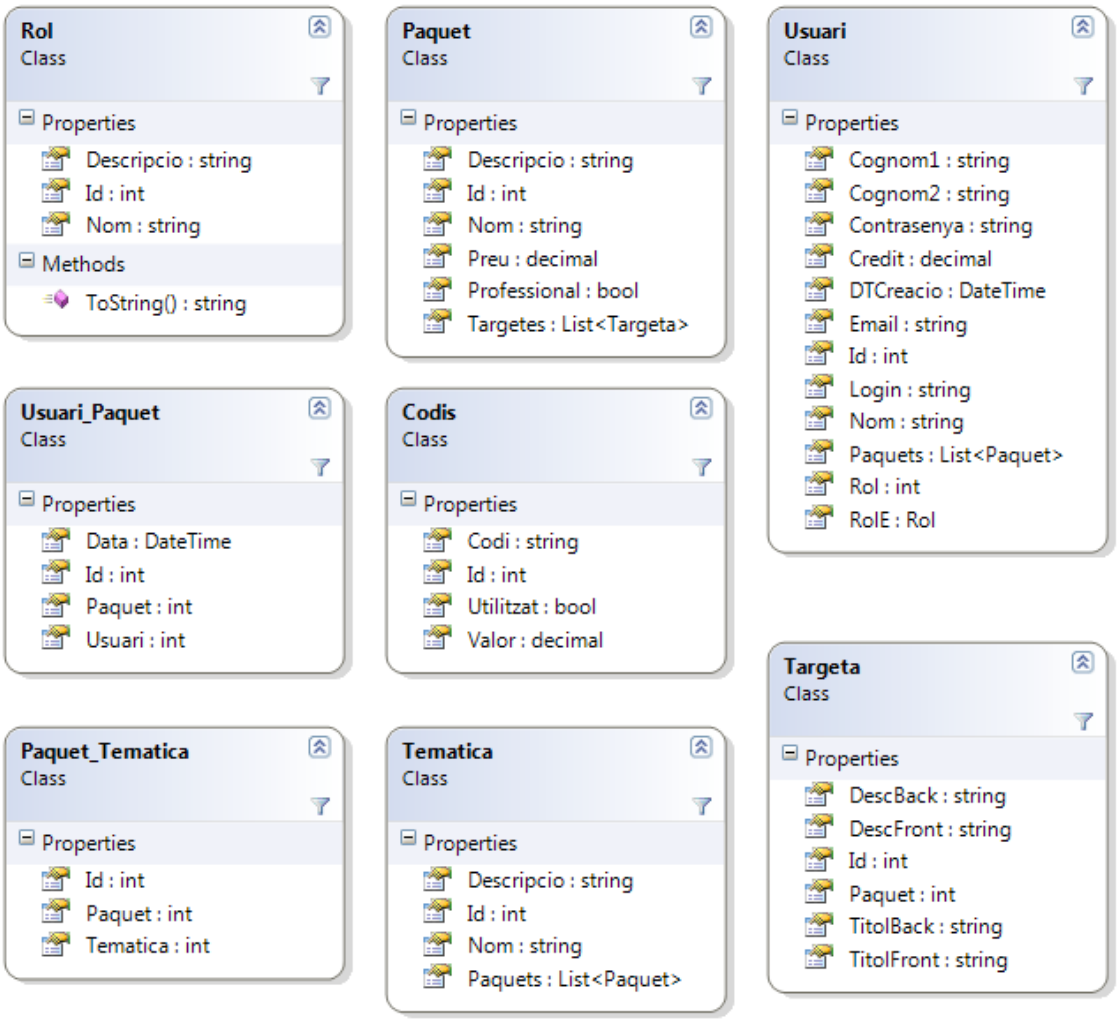
- **Resum**: L'administrador modifica les dades d'un usuari.
- **Actors**: Administrador.
- **Precondició**: L'usuari a modificar esta enregistrat al sistema.
- **Postcondició**: Les dades de l'usuari han estat modificades.
- **Descripció**: L'administrador accedirà a les dades de l'usuari i les modificarà quan sigui necessari reflectint els canvis als registres de la BBDD.

4.2.3.7.- Cas d'ús Eliminar usuaris

- **Resum**: L'administrador elimina un usuari.
- **Actors**: Administrador.
- **Precondició**: L'usuari a eliminar esta enregistrat al sistema.
- **Postcondició**: L'usuari serà eliminat del registre.
- **Descripció**: L'administrador eliminarà les dades del registre de l'usuari i aquest perdrà els drets d'utilitzar l'aplicació. Només s'eliminaran les dades quan l'usuari mateix ho demani o l'usuari infringeixi les normes d'ús de l'aplicació. L'usuari serà esborrat de la BBDD.

4.3.- Diagrama estàtic de disseny

Tot seguit és el diagrama estàtic de les classes de l'aplicació:



Com es pot comprovar des del diagrama estàtic original les classes principals han canviat ben poc. El canvi més gran que s'hi ha produït és l'adició d'una nova classe que és la de "Codis" ja que durant la implementació vaig adonar-me de que sense aquesta l'usuari estava condemnat a no poder tenir més diners ja que en aquell moment no hi havia cap sistema per a canviar el crèdit d'una persona; amb la classe codis això queda solucionat.

També hi ha hagut petits canvis en gairebé totes les classes per a poder (mitjançant Linq to SQL) emmagatzemar les dades que estan lligades a d'altres classes com ara la classe "Usuari" amb la classe "Rol" creant la propietat de classe *"RolE"*. Fent això ens estalviem l'haver de mirar el rol d'un usuari a la taula de rols cada cop que el volem saber, només cal que accedim a la seva propietat. Un exemple més clar encara és la creació de la propietat *"Targetes"* dins la classe "Paquet" per a poder emmagatzemar totes les instàncies de "Targeta" que pertanyessin a cada paquet.

Com es pot observar les classes de relació *N-N* entre taules com la "Paquet_Tematica" encara existeixen i encara que en la lògica de l'aplicació d'escriptori no es fan servir mai són fonamentals per a la lògica d'accés a dades.

També es pot veure que les classes principals són la de “Paquet”, “Targeta” i “Usuari”. Aquestes 3 classes són les que tenen tota la “carn” de l’aplicació. Sense aquestes classes el projecte quedaria en ben poc.

Una classe de suport però que és força important és la classe “Rol” ja que aquesta és l’encarregada de donar permisos a l’usuari com a administrador o usuari normal per a poder fer certes coses des de l’aplicació.

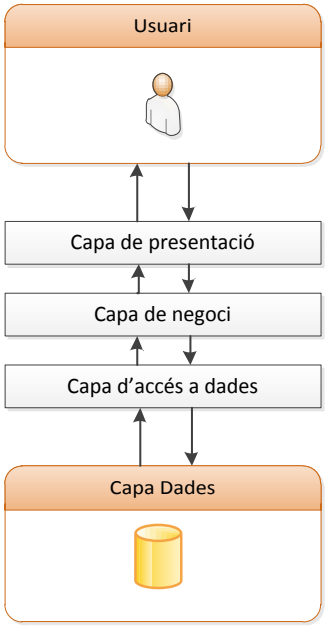
5.- Disseny de l'aplicació

5.1.- Arquitectura del sistema: Model de tres capes

Les característiques principals del model de tres capes són:

- La lògica del disseny i la lògica del negoci estan separades.
- S'utilitza una arquitectura escalable que facilita fer actualitzacions.
- Com que esta separat en diferents nivells, l'aplicació pot ser desenvolupada per diferents equips i cadascun només dedicar-se a la seva part.

En aquesta imatge podem veure com és una divisió per capes tradicional:

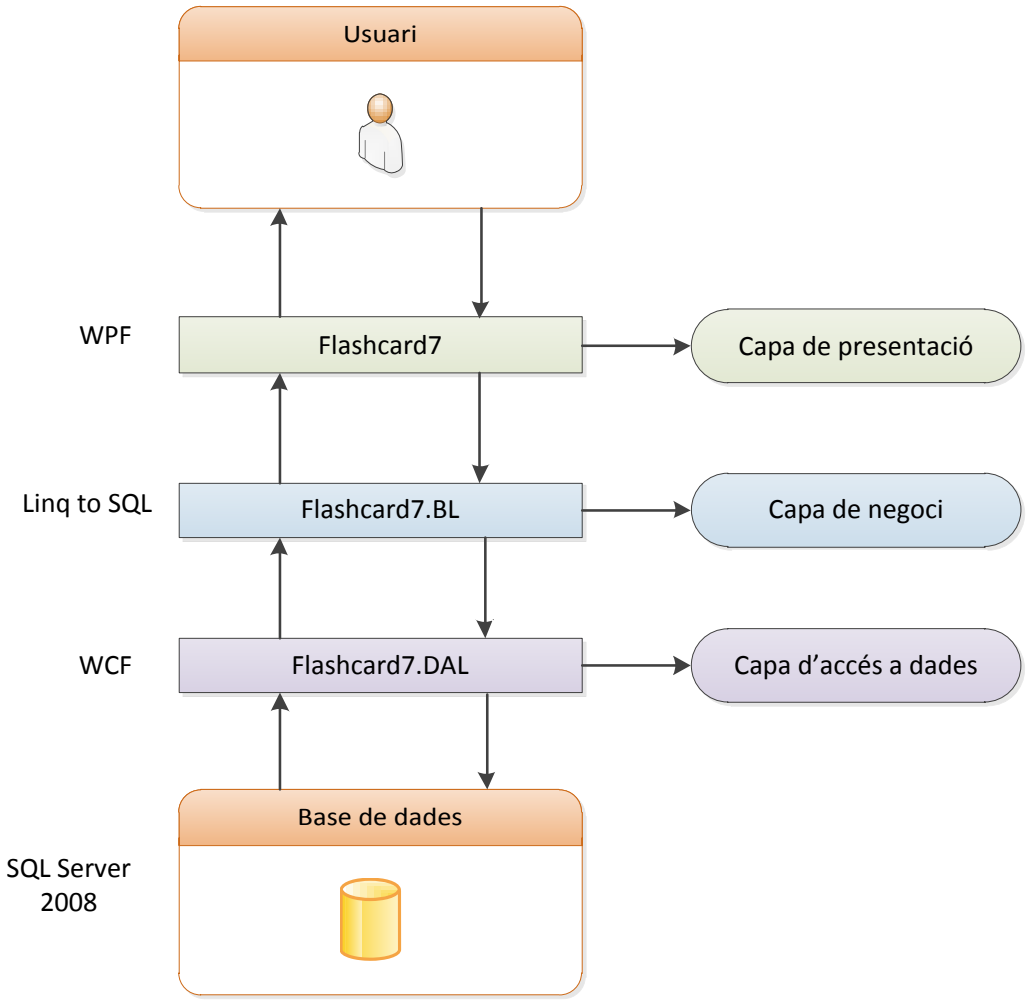


Aquestes capes són:

- **Capa de presentació**: És la que comunica l'usuari amb la capa de negoci, és la interfície gràfica de l'aplicació. Aquesta captura les dades i la informació que transmet l'usuari per a les altres capes i a la vegada retorna a l'usuari les dades que aquest ha demanat.
- **Capa de negoci**: En aquesta capa és on resideix tota la lògica i tots els processos de l'aplicació, es l'enllaç entre la capa de presentació i la capa de dades. Transmet i transforma la informació que es transmet entre aquestes capes.
- **Capa d'accés a dades**: La capa encarregada directa de extreure o emmagatzemar informació de la capa de dades. Aquesta capa retornarà a la capa de negoci la informació que aquesta li ha demanat o la introduirà a la capa de dades quan sigui necessari.

5.2.- Model de tres capes aplicat a l'aplicació

Aquest model de tres capes s'ha implementat a l'aplicació de la següent manera:



5.2.1.- WPF (Flashcard7, Aplicació)

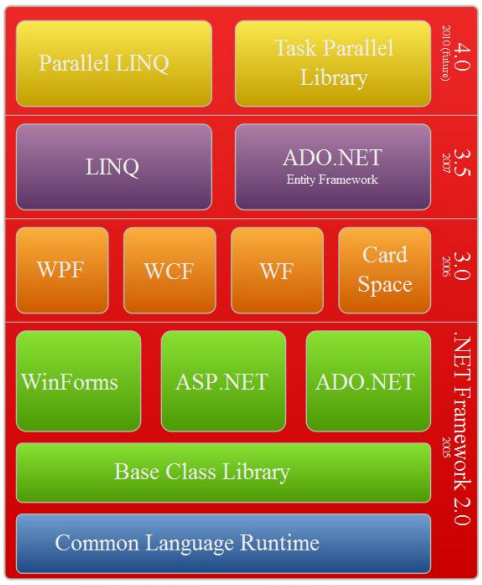
El WPF (*Windows Presentation Foundation)* és un subconjunt del marc de treball o *framework* .NET 4.0 (encara que hi és present des de la versió 3.0) i si s'han fet servir tecnologies com WinForms o ASP.NET ràpidament es poden veure les similituds. És un entorn que treballa directament amb DirectX i arrelat molt profundament a la presentació gràfica de les aplicacions ja que treballa amb un motor basat en càlculs vectorials.

El WPF es composa de dues parts, el llenguatge declaratiu de marcat o *markup*, XAML (*Extensible Application Markup Language*), i el codi subjacent o *code-behind*.

Per al desenvolupament d'aquesta part de l'aplicació s'ha utilitzat el Microsoft Expression Blend 4 per la part de XAML ja que aquesta eina està enfocada precisament en l'apartat de disseny de la interfície gràfica i per al *code-behind* s'ha combinat el mateix Expression Blend i el Visual Studio 2010.

5.2.2.- WCF (Flashcard7.DAL, Data Acces Layer)

El WCF (*Windows Communication Foundation*) és un kit de desenvolupament de programari per desenvolupar i desplegar serveis en sistemes Windows. Proveeix un entorn en temps d'execució per a serveis que permet exposar tipus CLR (*Common Language Runtime*) com a serveis i utilitzar altres serveis com a tipus CLR. WCF és una implementació de Microsoft per a

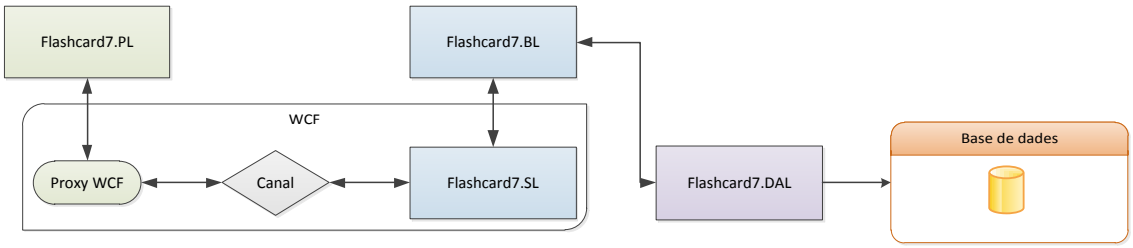


The .NET Framework Stack

un conjunt d'estàndards de la indústria definint interaccions entre serveis, conversions de tipus, càlcul de referències i la gestió de varis protocols.

Aquest kit és part del *framework* .NET 4.0 encarà que hi ha estat present des del .NET 3.0. L'actual és la tercera revisió del WCF i des de l'original de l'any 2006 s'hi ha afegit eines per a la gestió de instàncies de serveis, crides asíncrones, gestió de transaccions, més seguretat i d'altres.

Aquesta capa entra en joc quan des de la capa de presentació s'envia una sol·licitud de dades al servei WCF. La capa de negoci processarà aquesta sol·licitud a través dels seus mecanismes fins a la capa de serveis mitjançant un *proxy* i un canal de dades traduint la informació per a que la següent capa ho entengui. El *proxy* serà l'actual interceptor de les crides i s'abstraurà de la lògica necessària per a connectar-se amb el servei que es vol consumir i el canal serà el mitjà per on el missatge viatjarà entre el client i l'ens d'acollida del servei. Aquesta traducció de la informació es fa amb les eines del WCF, les del .NET 4.0 i les API del sistema operatiu, i en conjunt creen una resposta que es retorna altre cop a la capa de presentació.



5.2.3.- .NET (Solució Flashcard7)

El .NET 4.0 és un marc de treball que inclou una gran llibreria i recolza diferents llenguatges de programació que permeten interoperabilitat entre ells. La llibreria .NET està disponible a tots els llenguatges de programació que el .NET suporta.

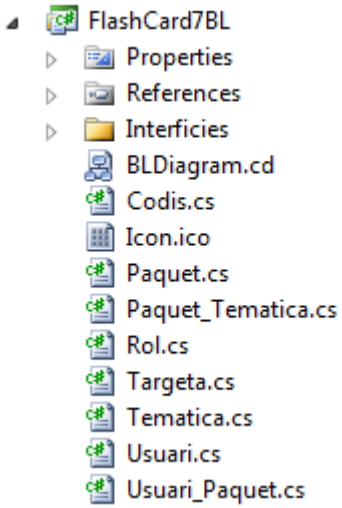
La llibreria de classes base del *framework* proveeix interfícies d'usuari, accés a dades, connectivitat amb bases de dades, criptografia, desenvolupament d'aplicacions web, algoritmes numèrics, i comunicació en xarxa. El programador és l'encarregat de barrejar aquestes llibreries amb el seu propi codi per crear les aplicacions.

Els programes escrits per el marc de treball .NET s'executen en l'entorn CLR (*Common Language Runtime*). El CLR és una maquina virtual d'aplicacions per que els programadors no hagin de tenir en compte quin processador executarà el programa. El CLR també proporciona altres serveis importants com la seguretat, la gestió de memòria i la manipulació d'excepcions. La llibreria de classes i el CLR constitueixen el marc de treball .NET.

L'entorn de treball Visual Studio és l'eina ideal per a treballar amb aquest marc de treball gràcies a totes les ajudes que proporciona un entorn de treball dedicat com aquest.

Amb totes les eines de .NET 4.0 crearem un conjunt d'entitats que seran les encarregades de fer de mitjà de comunicació entre la capa de serveis i la capa d'accés a dades.

En la següent imatge es pot veure les classes que composaran aquesta capa dins el projecte FlashCard7BL.



5.2.4.- Linq to SQL (Flashcard7.BL, Business Layer)

El Linq (*Language Integrated Query*) és un component del .NET que afegeix capacitats natives de consulta de dades als llenguatges de .NET.

El Linq defineix un conjunt de noms de mètodes anomenats *standard query operators* (operadors de consultes estàndard) o *standard sequence operators* (operadors de seqüència estàndard), mitjançant regles de traducció de les anomenades expressions de consulta a expressions que usen aquests noms de mètodes, expressions lambda i tipus anònims. Aquests poden, per exemple, ser usats per projectar i filtrar dades en *arrays*, classes enumerables, XML, bases de dades i altres fonts de dades.

El Linq proveeix una sintaxi i una semàntica molt semblant al SQL, i amb moltes de les seves avantatges. Amb Linq es pot canviar l'estructura subjacent de les dades que estan sent consultades sense la necessitat de fer cap canvi en el codi que finalment fa la consulta. Tot i això, s'ha de tenir en compte que encara que s'assembli tant al SQL, el Linq és molt més flexible i pot manipular una varietat més gran d'estructures lògiques de dades.

El proveïdor *Linq to SQL* permet que el Linq sigui utilitzat per a fer consultes a bases de dades del SQL Server. Com que les dades del SQL Server normalment estan en un servidor remot, i com que el SQL Server té el seu propi motor de consultes, Linq to SQL no utilitza el mateix motor de consultes que el Linq; en canvi, converteix un consulta Linq a una consulta SQL que és enviada al SQL Server per al seu processament. Però, com que el SQL Server guarda les dades com a dades relacionals i el Linq treballa amb dades encapsulades en objectes, les dues representacions han de ser "mapejades" les unes a les altres. Per aquesta mateixa raó, Linq to SQL també defineix un marc de treball de mapejat. Aquest es fa definint classes que

corresponen a les taules de la base de dades i que contenen totes o un subconjunt de columnes de la taula com a membres de dades. La correspondència, juntament amb altres atributs del model relacional com les claus primàries, estan especificades utilitzant atributs definits per el Linq to SQL.

Per exemple amb la classe Paquet:

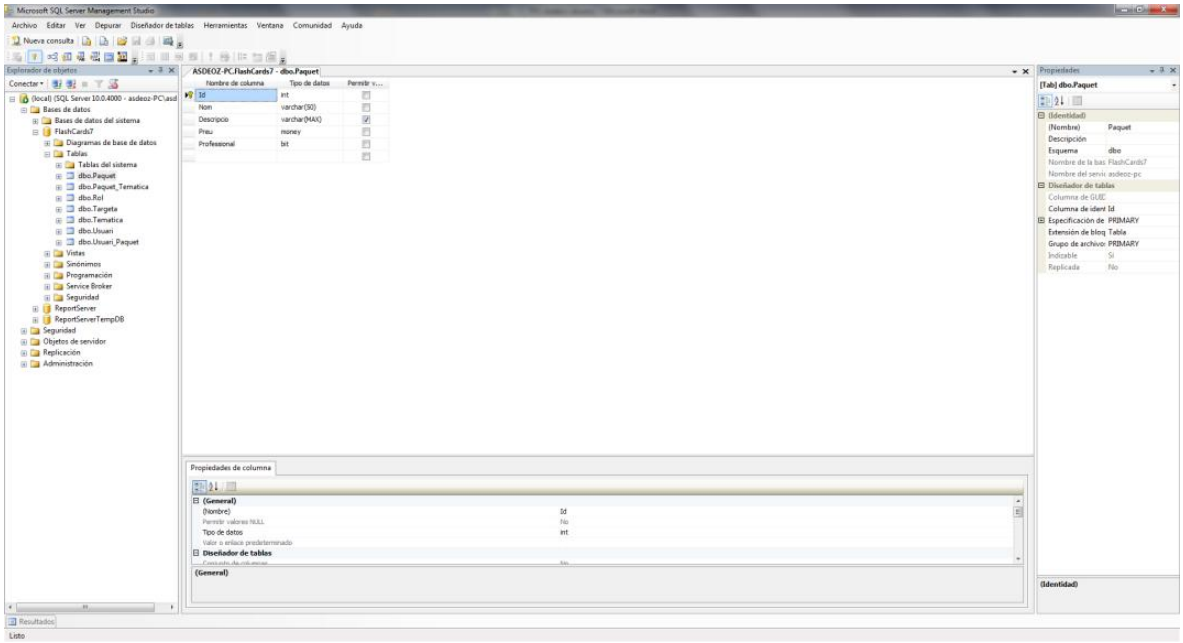
```
[Table(Name = "Paquet")]
[DataContract]
public class Paquet
{
    private List<Targeta> _targetes;

    [Column(IsPrimaryKey = true, CanBeNull = false, IsDbGenerated = true)]
    [DataMember]
    public int Id;

    [Column(CanBeNull = false)]
    [DataMember]
    public string Nom;

    [Column(CanBeNull = true)]
    [DataMember]
    public string Descripcio;

    [Column(CanBeNull = false, DbType = "money")]
    [DataMember]
    public decimal Preu;

    [Column(CanBeNull = false)]
    [DataMember]
    public bool Professional;

    [DataMember]
    public List<Targeta> Targetes;
}
```

Aquesta definició de classe està mapejada a la taula Paquet i cada membre es correspon amb una columna. Les classes han d'estar definides abans que el Linq to SQL pugui ser utilitzat.

5.2.5.- SQL Server 2008 (FlashCards7.dbo)

El SQL Server 2008 és un SGBD (Sistema Gestor de Bases de Dades relacionals) basat en el llenguatge SQL. És una eina capaç de proporcionar grans quantitats de dades a molts usuaris simultàniament.

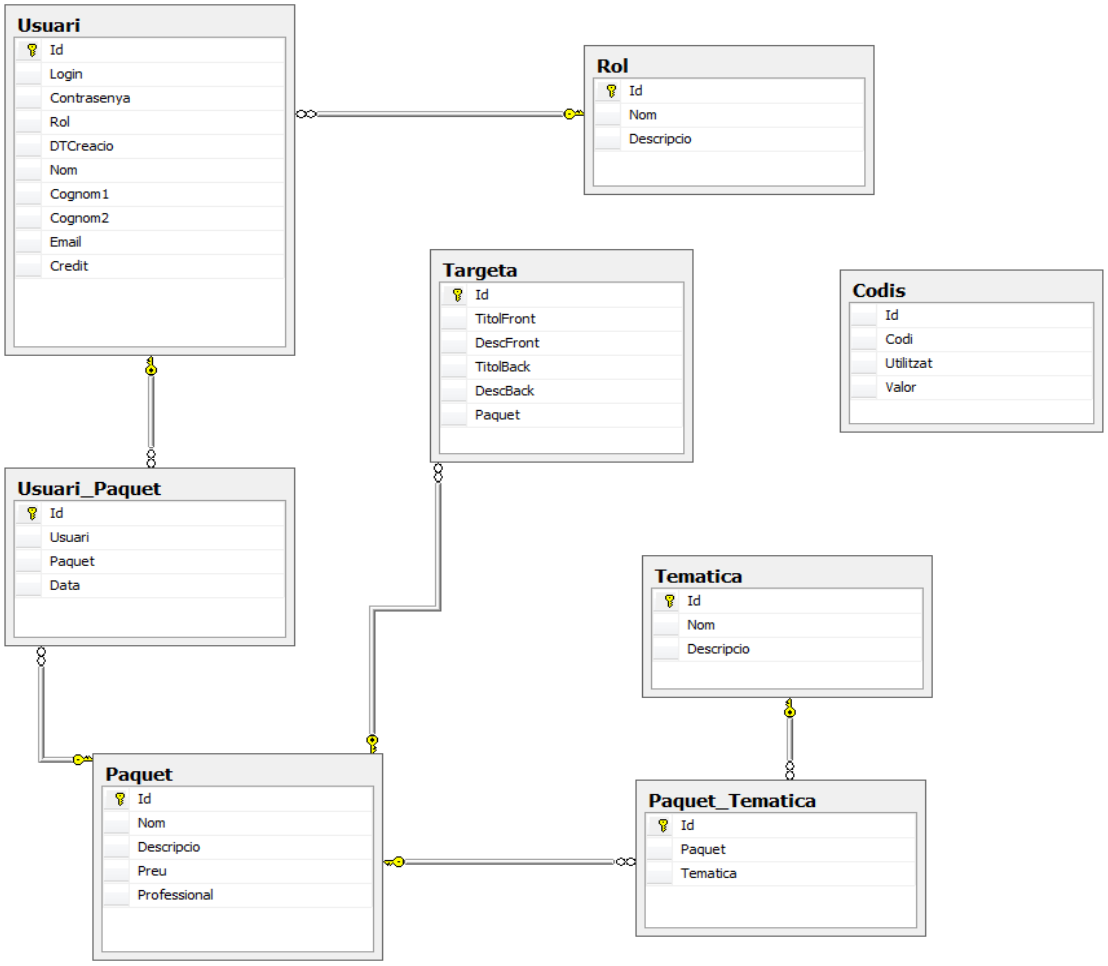
Per a administrar el SQL Server 2008 utilitzarem una eina integrada que ens ofereix un entorn gràfic i un conjunt d'eines força simples i potents per a crear i modificar convenientment la base de dades anomenat SQL Server Management Studio.



5.3.- Base de Dades FlashCards7.dbo

5.3.1.- Diagrama de la Base de Dades

En la següent imatge es pot veure el disseny de la base de dades:



5.3.2.- Descripció de la Base de Dades

En el següent quadre hi ha una definició de la base de dades taula a taula.

Taula	Descripció
Rol	La clau principal de la taula és *Id*, un número identificador únic. El **Rol** serveix per saber, per a un **Usuari**, quin grau d'accés té a l'aplicació. Els camps són auto-explicatius.
Usuari	La clau principal de la taula és *Id*, un número identificador únic. L'**Usuari** té una clau forana ***Rol*** que indica els permisos d'aquest usuari. Els camps *Nom*, *Cognom1*, *Cognom2* i *Email* es refereixen a les dades reals darrere un usuari. El *Login* i la *Contrasenya* són el nom i mot de pas que utilitzarà un usuari per identificar-se al sistema. El camp *DTCreacio* es la data de creació del compte. El camp *Credit* es refereix als diners que l'usuari té en el compte per a poder fer compres en el mercat de l'aplicació.
Paquet	La clau principal de la taula és *Id*, un número identificador únic. Els camps *Nom* i *Descripcio* són camps auto-explanatoris. El camp *Professional* és un camp booleà que defineix si aquest paquet l'ha creat un usuari o un administrador. El camp *Preu* tindrà valor si el paquet l'ha creat un administrador i aquest l'ha posat al mercat.
Usuari_Paquet	La clau principal de la taula és *Id*, un número identificador únic. Com que la relació entre **Usuari** i **Paquet** és una relació N-N, **Usuari_Paquet** és una taula intermèdia. Els camps ***Usuari*** i ***Paquet*** són claus foranes a les seves respectives taules. El camp *Data* és la data en que aquesta relació es va crear.
Targeta	La clau principal de la taula és *Id*, un número identificador únic. La **Targeta** representa una targeta d'un paquet, per això el camp ***Paquet*** fa de clau forana cap a aquest i ens indica a quin **Paquet** pertany. Els camps *TitolFront* i *DescFront* són el títol i la descripció de la part davantera d'una targeta mentre que els camps *TitolBack* i *DescBack* són el títol i la descripció de la part de darrera.
Tematica	La clau principal de la taula és *Id*, un número identificador únic. La **Tematica** és principalment el tema del que tracten els paquets de targetes i pel camp dels paquets pel qual trobarem paquets en el cercador. Els camps són auto-explanatoris.
Paquet_Tematica	La clau principal de la taula és *Id*, un número identificador únic. Com que la relació entre **Paquet** i **Tematica** és una relació N-N, **Paquet_Tematica** és una taula intermèdia. Els camps ***Paquet*** i ***Tematica*** són claus foranes a les seves respectives taules.
Codis	La clau principal de la taula és *Id*, un número identificador únic. El camp *Codi* és un codi de 25 caràcters que serà el que l'usuari tindrà, *Valor* es refereix a el valor en euros de la targeta i *Utilitzat* és un camp booleà que canviarà quan el codi hagi estat fet servir.

*Les paraules en **negreta** mostren noms de taules, les paraules en *cursiva* mostren noms de columnes i les paraules amb els dos ***trets*** mostren noms de columnes que són claus foranes a altres taules.

L'únic canvi des del primer disseny de la base de dades ha estat la inclusió d'una nova taula anomenada **Codis** per a poder tenir en un sistema d'emmagatzematge persistent els codis de

les targetes per a poder incrementar el crèdit dels usuaris i saber quan estat utilitzades i quan no.

Els canvis que s'han hagut de fer han fet un impacte mínim sobre el disseny de les classes de l'aplicació ja que el Linq to SQL ajuda força i la única taula que s'ha inclòs no tenia cap relació amb les altres.

6.- Interfície gràfica

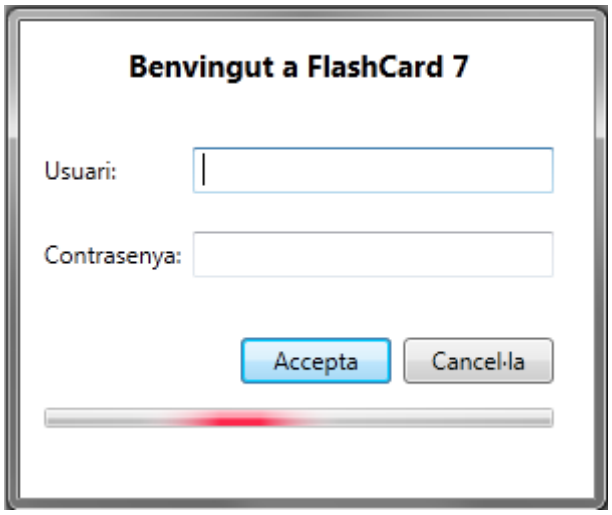
L'apartat gràfic de l'aplicació s'ha desenvolupat amb la tecnologia WPF utilitzant els entorns de treball Visual Studio 2010 i Microsoft Expression Blend 4.

WPF està basat en el llenguatge declaratiu XAML i permet crear entorns gràfics molt rics que alhora permeten una navegació agradable per a l'usuari.

6.1.- Aplicació d'escriptori

A continuació podrem observar uns quants apartats de l'aplicació d'escriptori.

6.1.1.- Login



El primer que veu l'usuari quan comença l'aplicació és la finestra de benvinguda o *login*. En aquesta finestra l'usuari introduirà el seu nom d'usuari i la seva contrasenya i es validarà contra la base de dades per a poder accedir a les funcionalitats de l'aplicació.

Tant el nom d'usuari com la contrasenya són valors que l'usuari haurà decidit a l'hora de registrar-se.

Un cop validat l'usuari rebrà un rol que li donarà més o menys accés a les funcionalitats de l'aplicació i decidirà si és un usuari normal o un administrador.

6.1.2.- Menú principal

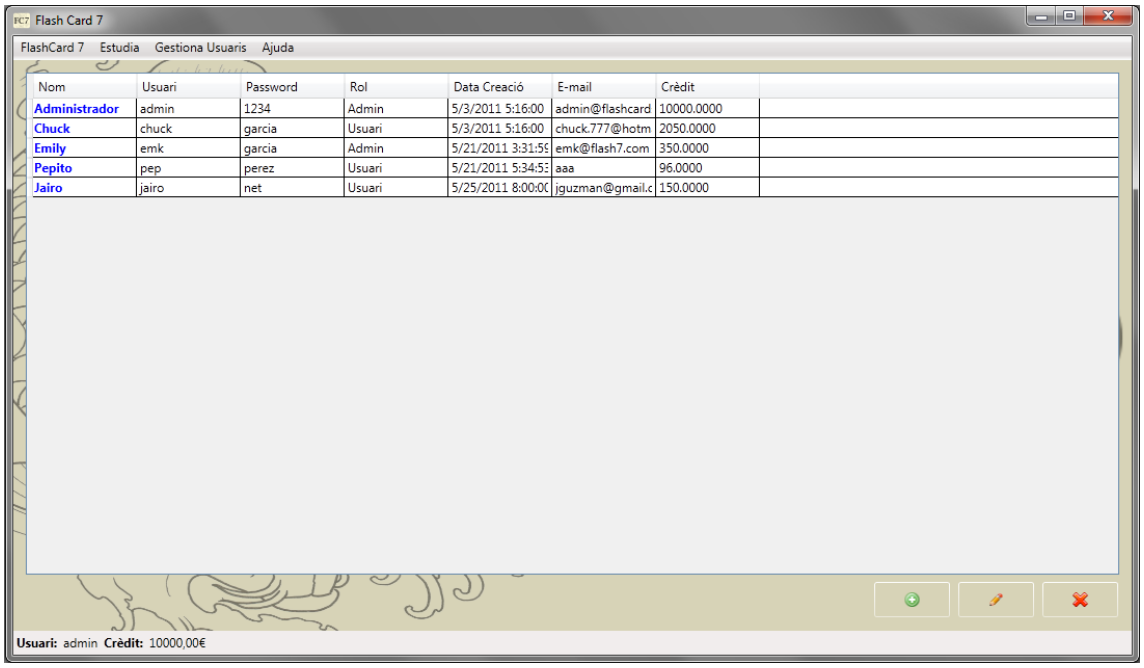
Un cop l'usuari s'ha validat, el menú principal serà la següent finestra que s'obrirà. En aquesta finestra tindrem, a primera vista a la part de baix a l'esquerra, el nom d'usuari i el crèdit.

A la part superior hi trobarem la barra d'eines des d'on accedirem a tots els altres apartats de l'aplicació. La part de gestionar usuaris només estarà disponible pels administradors.

Menú principal

6.1.3.- Gestió d'usuaris

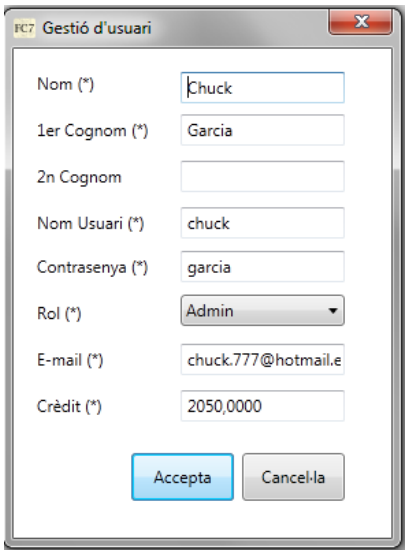
Si l'usuari s'ha validat com a administrador llavors aquest podrà accedir a l'apartat de gestió d'usuaris.



Gestió d'usuaris

Des d'aquí i mitjançant un simple sistema de control l'administrador podrà accedir a totes les dades dels usuaris per a poder fer qualsevol modificació necessària.

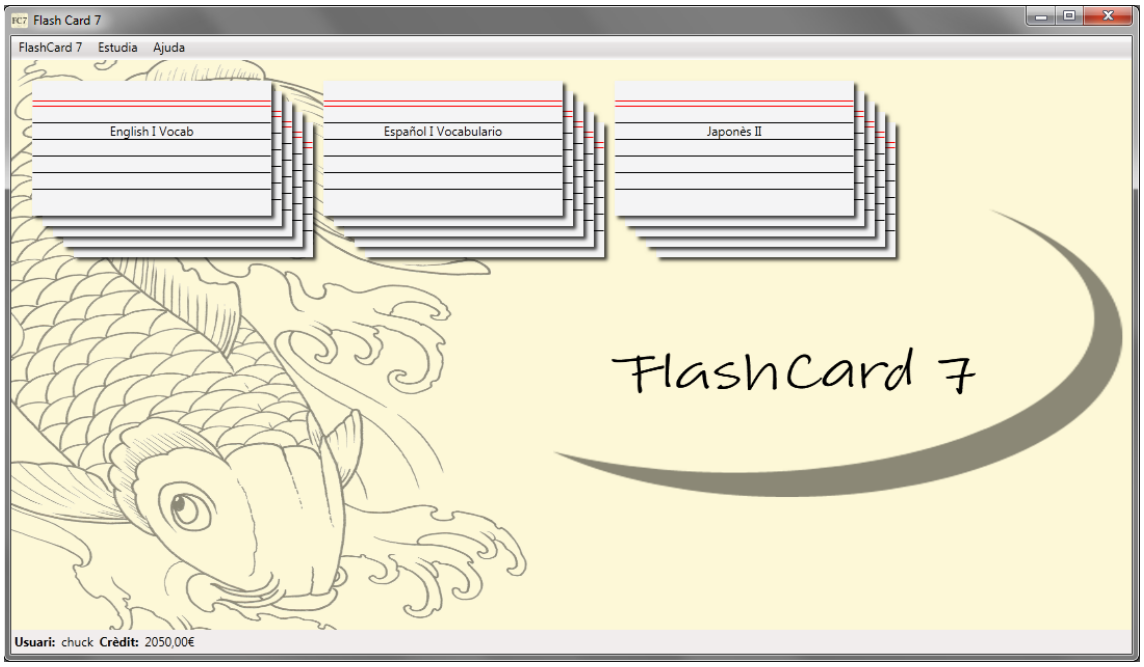
Els usuaris normals (de moment) per a poder modificar el seu compte s'hauran de posar en contacte amb l'administrador per que aquests facin els canvis pertinents.



Quan pitgem el botó d'afegir usuari o el botó de modificar un usuari ens apareix una finestra per a poder posar les dades d'aquest. Un cop s'ha emplenat els quadres amb la informació corresponent al prémer el botó d'acceptar es crearà l'usuari o se'n modificaran les dades.

6.1.4.- Menú de paquets de targetes

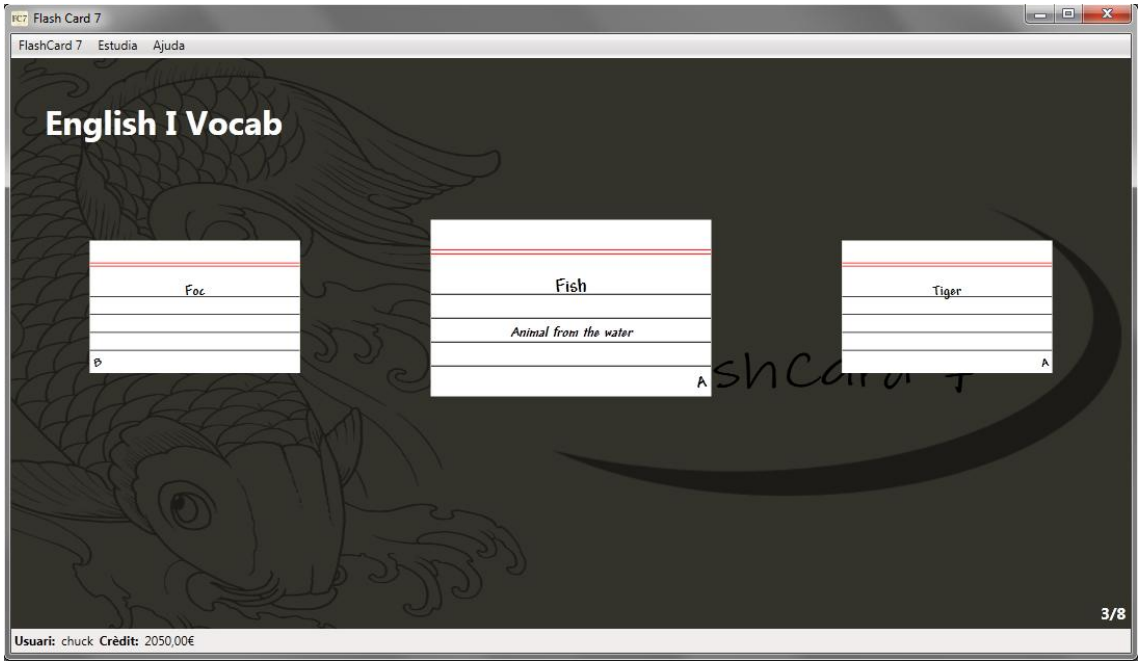
Des del menú de paquets de targetes l'usuari podrà accedir de forma ràpida i fàcil als seus paquets per poder començar a estudiar amb les targetes.



Menú de paquets de targetes

A primera vista apareixerà un menú com el de la imatge amb tants quadres com paquets tingui l'usuari amb el nom del paquet dins del quadre. Només caldrà trobar el paquet i pressionar-lo per a accedir a la zona d'estudi i començar a fer servir les targetes.

6.1.5.- Zona d'estudi



Zona d'estudi

Aquesta és la part més important de l'aplicació ja que, al cap i a la fi, és una aplicació per a estudiar. Aquesta finestra serà una finestra molt simple per què l'usuari no es distregui per qualsevol cosa. A la part superior es pot veure sempre el nom del paquet que s'està fent servir mentre que al centre de la pantalla només si mostren tres targetes. Les dues de les bandes són l'anterior i la següent que van correlatives a la central i es pot accedir a elles mitjançant el ratolí. La targeta central és la que s'està utilitzant en aquest moment i com es pot veure hi ha el títol i la descripció frontal de la targeta i, si es clica en ella, la targeta es gira mostrant-nos la solució del que es planteja davant.

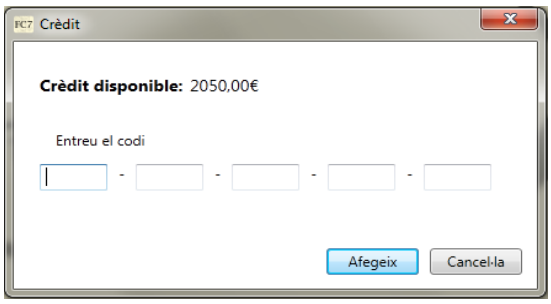
Per a saber quina cara s'està mostrant en aquest moment només cal mirar a la cantonada de la targeta i si hi veiem una "A" és que és la cara de davant i si hi ha una "B" és la de darrere.

A la pila de l'esquerra les cartes sempre quedaran girades ja que estem simulant una pila de cartes que ja hem fet servir.

En la part inferior dreta sempre podem veure el nombre de targeta actual i el total dins del paquet per sempre saber en quin punt de l'estudi estem.

Com es pot veure s'ha fet aquest apartat el més simple possible per a facilitar l'estudi a l'usuari.

6.1.6.- Afegir crèdit



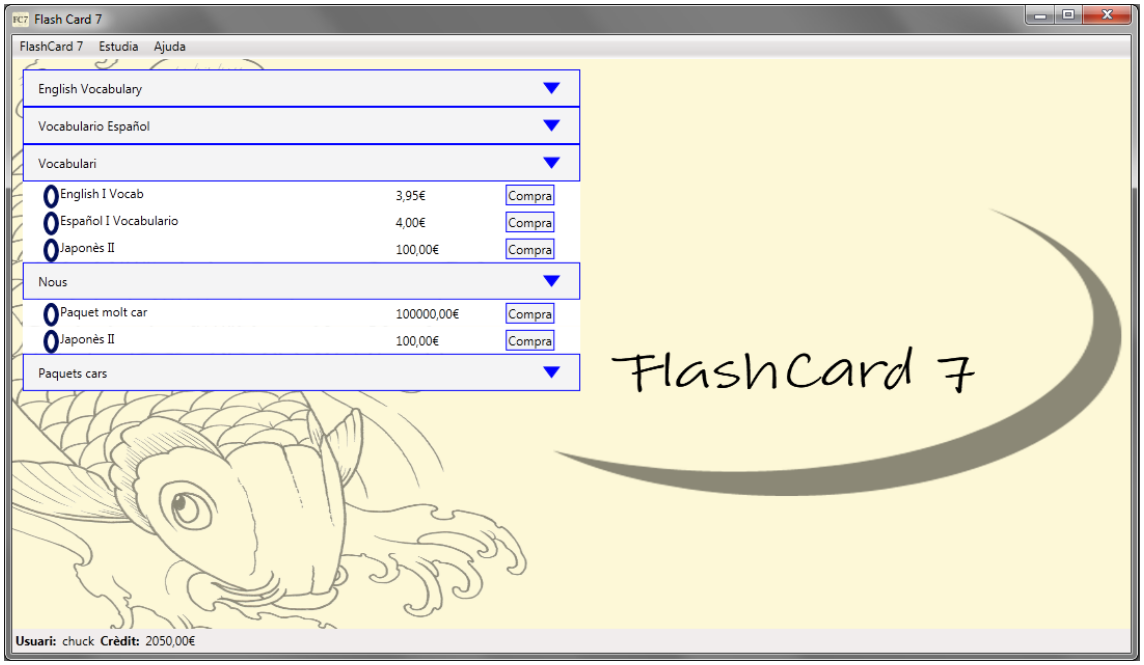
Com tot en aquest món les targetes professionals també s'han de pagar i per a fer-ho de manera fàcil s'ha creat aquest menú on, introduint un codi de 25 caràcters el qual es pot aconseguir mitjançant la plana web o en

tendes especialitzades, es pot afegir crèdit al saldo de l'usuari.

El codi es validarà contra el sistema i un cop validat s'afegirà crèdit al compte per a poder fer tots els pagaments sense necessitat de fer servir una targeta de crèdit cada cop.

6.1.7.- Mercat de paquets

Per últim però no per això menys important, està el mercat on l'usuari podrà comprar paquets de targetes per a poder utilitzar després.



Mercat de paquets

Els paquets estaran ordenats per temàtiques a la que pertanyen en pestanyes tal com es veu en la imatge. Si fem click en la pestanya es desplegarà una llista amb tots els paquets que pertanyen a aquesta temàtica amb els seus preus i un botó amb la opció de compra.

Un cop s'ha trobat el paquet desitjat, i si es té el crèdit necessari, es pot prémer el botó "Compra" i el paquet ja passarà a formar part de la col·lecció de l'usuari i ja estarà a punt per a poder-lo utilitzar per a estudiar.

Si per equivocació l'usuari premés en un paquet que ja té no passaria res ja que el sistema ja l'avisaria de que aquest paquet ja forma part de la seva col·lecció.

7.- Objectius aconseguits

Del que es va proposar al principi al resultat final hi ha hagut algunes diferències, sobretot en l'apartat de l'aplicació per a dispositius mòbils.

Començarem amb les diferències que hi ha hagut entre els objectius no aconseguits i els aconseguits per a l'aplicació d'escriptori.

Per començar una de les grans diferències és la possibilitat de crear targetes personals... Aquesta opció va ser una de les que, per falta de temps, vaig haver de treure de l'aplicació. M'hauria agradat molt poder-la incloure però no vaig preveure que les corbes d'aprenentatge d'algunes tecnologies que havia d'utilitzar eren molt més llargues del que m'imaginava i tampoc vaig tenir prou en compte que no només era una nova tecnologia si no que n'eren moltes. Per aquesta raó aquesta possibilitat va ser eliminada de la versió final.

Per altra part una altra de les diferències va ser la possibilitat de treballar *offline*. Aquesta és una opció que vaig treure expressament ja que crec que deixar que l'usuari accedís a l'aplicació sense fer el procés de *login* contra el servidor principal podria ser una falla en la seguretat de tot el sistema i podria comprometre la integritat del producte.

Una de les funcions que s'han millorat respecte al proposat en un principi és l'apartat gràfic. Aquesta poder no és una de les funcions útils que es poden notar en forma de servei però vaig trobar oportú dedicar-hi una mica més de temps ja que crec que aquesta és una aplicació on l'aspecte visual i el *feeling* que ha de retornar a l'usuari ha de ser atractiu per a que aquest es senti confortable utilitzant aquesta eina com a utilitat habitual per a l'estudi i el dia a dia.

Sobre l'apartat del servei web tot més o menys ha quedat com es va proposar des de bon principi, molt poques coses, per no dir gairebé res, han canviat.

I finalment, un dels principals objectius que no s'ha aconseguit ha estat l'aplicació per a dispositius mòbils. Encara que es va fer l'estudi sobre les tecnologies i intentar buscar informació a tot arreu no vaig aconseguir que l'aplicació mòbil pogués fer ni una connexió exitosa amb el servei web i això va fer que aquest apartat del projecte quedés totalment anul·lat. Encara ara no sé quin és el problema i perquè una cosa que hauria de ser fàcil no ha tingut solució. El meu consultor és un testimoni de que no s'hi va poder fer res tot i tenir temps.

8.- Treball futur

Aquesta aplicació està pensada per a poder implementar moltes altres funcionalitats i en un futur poder-se convertir en una eina molt útil per a molta gent. Estic gairebé segur que aquest programa es podria millorar en algunes de les següents àrees:

- Un apartat on l'usuari pugui crear les seves pròpies targetes i paquets i els pugui pujar a la xarxa per poder-hi accedir amb qualsevol dispositiu que tingui l'aplicació instal·lada i des de qualsevol lloc.
- Si l'usuari pogués crear els seus paquets deixar-li, fins i tot, compartir alguns d'aquests (amb algunes limitacions o cobrant alguna cosa pel servei) entre els seus amics o companys.
- Per suposat crear l'aplicació mòbil per a que l'usuari pugui accedir des de qualsevol lloc amb el seu *smartphone* amb connectivitat a internet.
- Crear una pàgina web on els usuaris puguin entrar i gestionar la seva compte i comprar targetes de prepagament.
- Crear un apartat nou on, l'usuari que vulgui, pugui eliminar paquets de la seva compte.

Segurament es podrien trobar encara moltes més opcions i millores ja que crec que aquesta aplicació podria tenir moltes possibilitats en el mercat.

9.- Conclusions

El primer sentiment que tinc al fer una repassada a tot el projecte és d'haver proposat molta més feina de la que, pel temps i els meus coneixements anteriors, es podien assolir.

Puc dir per això, que fent el TFC m'he sentit realment passant per totes les fases d'un projecte com el que em podria trobar davant en qualsevol moment de la meva vida laboral. Ha sigut una tasca molt enriquidora, de principi a fi no he parat de descobrir i aprendre coses noves que de ben segur em seran molt útils.

Quan vaig començar el projecte jo ja havia fet una mica de programació en C# i tenia, el que creia que era, uns bons fonaments utilitzant el Visual Studio. Poder per aquestes raons em vaig animar i vaig proposar un projecte una mica massa gran, ara veig que hauria necessitat més del doble d'hores de les que hi he pogut dedicar.

Els primers problemes van ser al passar de *WinForms* a WPF, al principi no em semblava que hi hagués una diferència gaire gran però ben aviat em vaig endur una sorpresa agredolça, per una part això suposava dedicar més hores de les que hi volia dedicar a canviar de tecnologia, però per altra part vaig veure que les possibilitats gràfiques amb WPF eren molt més grans i això em va agradar.

Un cop passat aquest primer tràngol em vaig trobar amb altres tecnologies que tampoc havia tocat mai com eren WCF, Silverlight o Linq to SQL. Una a una han sigut unes experiències molt enriquidores però totes a la vegada a vegades m'ha sembla que m'havia de tornar boig i veia com el temps se'm tirava a sobre: em sembla que mai en la vida m'havia passat tantes hores seguides davant l'ordinador com em vaig passar durant la fase d'implementació provant coses i més coses fins que els ulls ja se'm tancaven sols.

Petites coses sempre es posen al mig com quan te'n adones que després de passar 5 hores fent una cosa hi ha una manera molt més ràpida i fàcil de fer-la però clar, llavors ja és massa tard, però el més frustrant va ser a l'arribar a fer la implementació de l'aplicació mòbil. Després de dedicar hores i hores aprenent Silverlight, fent proves, creant un bon disseny per a l'aplicació per què sigui fàcil d'utilitzar i clara te n'adones que, per un problema que no saps d'on prové, no ets ni capaç de fer que un usuari faci *login* al sistema i que ningú et pot ajudar, llavors és quan te'n duus un gran disgust. Em sembla que he passat gairebé la mateixa quantitat d'hores fent gran part de l'aplicació d'escriptori i el servei web com intentant que l'aplicació mòbil fes alguna cosa.

Tot i això la conclusió general del TFC ha sigut molt bona, la quantitat de coses que he après en aquests mesos ha sigut increïble i m'han donat uns bons fonaments per a voler-me endinsar encara més en el que és aquest món del .NET de Microsoft. Ara mateix estic segur que si hagués de fer l'aplicació podria millorar moltes parts del codi i moltes de les funcionalitats fer-les millor però això només mostra un cop més que alguna cosa he après de tot això.

10.- Bibliografia

10.1.- Llibres

- **Programming WCF Services 3rd Edition**, *Juval Löwy*, 2010 Ed. O'Reilly
- **Microsoft Visual C# 2010 Step by Step**, *John Sharp*, 2010 Ed. Microsoft Press
- **Silverlight 4 Step by Step**, *Laurence Moroney*, 2010 Ed. Microsoft Press
- **Build a Program Now! Visual C# 2008 Express Edition**, *Patrice Pelland*, 2008 Ed. Microsoft Press
- **Programming C# 3.0**, *Jesse Liberty & Donald Xie*, 2008 Ed. O'Reilly
- **Microsoft Expression Blend 4 Step by Step**, *Elena Kosinska and Chris Leeds*, 2011 Ed. Microsoft Press
- **Visual Studio 2010 and .NET 4.0 Six-in-One**, *István Novák & András Velvárt & Adam Granic & György Balássy & Attila Hajdrik & Mitchel Sellers & Gastón C. Hillar & Ágnes Molnár & Joydip Kanjilal*, 2010 Ed. Wiley Publishing
- **WPF 4 Unleashed**, *Adam Nathan*, 2010 Ed. SAMS

10.2.- Recursos online

- http://www.wpftutorial.net/
- http://www.elguille.info/
- http://msdn.microsoft.com/
- http://en.wikipedia.org/

10.3.- Recursos humans

Donar especial gràcies a gent que m'ha ajudat a resoldre dubtes i a fer possible el projecte: Rob Miles des de Twitter com a @robmiles, John Sharp que molt amablement em va ajudar via e-mail amb WCF i al consultor Jairo Sarrias que em va ajudar fins l'últim moment en qualsevol dubte que li presentés i em va poder donar més temps quan van sorgir problemes grans.