

## **Aplicació web per a un casino online**



**Víctor Martín Pereña**

Enginyeria Tècnica en Informàtica de Gestió

**Consultor: Jordi Sánchez Cano**

11/06/2010

# Índex de continguts

<b>1. Introducció .....</b>	<b>3</b>
1.1. Justificació del projecte.....	3
1.2. Objectius del projecte.....	4
1.3. Planificació del projecte .....	4
1.4. Productes obtinguts .....	6
<b>2. Anàlisi .....</b>	<b>7</b>
2.1. Anàlisi de requeriments funcionals .....	7
2.1.2. Subsistema de partida .....	7
2.2. Anàlisi de requeriments tècnics.....	8
2.4. Casos d'ús .....	10
2.4.1. Actors .....	10
2.4.2. Diagrames de casos d'ús.....	11
2.4.2. Descripció dels casos d'ús .....	13
<b>3. Disseny Tècnic.....</b>	<b>19</b>
3.1. Disseny de l'arquitectura de l'aplicació.....	19
3.2. Disseny de les classes .....	21
3.3. Diagrama de classes .....	22
3.4. Fitxes CRC de les classes .....	23
3.4.1. Classes d'entitat .....	23
3.4.2. Classes de comunicació .....	24
3.5. Diagrames de col·laboració .....	26
3.5.1. Diagrama de col·laboració del cas d'ús Alta Partida.....	26
3.5.2. Diagrama de col·laboració del subsistema Partida .....	27
3.6. Diagrames d'activitats .....	28
3.6.1. Diagrama d'activitats relacionades amb l'alta d'un usuari .....	28
3.7. Disseny de la interfície gràfica d'usuari .....	29
3.7.1. Diagrama de flux de les pantalles.....	29
3.8. Disseny de la base de dades.....	30
3.8.1. Descripció de les taules .....	30
3.7.2. Diagrama de la base de dades .....	31
<b>4. Implementació .....</b>	<b>32</b>
4.1. Modificacions respecte al disseny .....	32
4.1.1. Variacions a les classes d'entitat .....	32
4.1.2. Variacions a les classes de comunicació i model de dades.....	33
4.1.3. Variacions a la base de dades .....	33
4.2. Interfície gràfica .....	35
<b>5. Línies de treball futur .....</b>	<b>44</b>
<b>6. Conclusions .....</b>	<b>44</b>
<b>7. Glossari .....</b>	<b>45</b>
<b>8. Bibliografia .....</b>	<b>47</b>

# 1. Introducció

## 1.1. Justificació del projecte

Durant els darrers mesos el pòquer s'ha estat expandint més que mai, i lluny de quedar-se tancat dins l'àmbit dels casinos, ha aconseguit introduir-se de manera virtual a moltes llars mitjançant els recursos que ofereix Internet. No obstant, la major part d'aplicacions oferides arreu de la xarxa no s'ofereixen com a aplicacions web basades en el protocol http o https, sinó més comunament, com a aplicacions que s'han d'instal·lar a la màquina client, i es connecten a un servidor mitjançant sockets. És aquí on rau la justificació d'aquest treball, juntament amb els objectius del projecte.

Donada la quantitat de variants del pòquer, i amb l'objectiu d'acotar l'abast del projecte, s'ha triat només una de les modalitats: el Texas hold 'em sense límit d'apostes.

Els requeriments funcionals previstos de l'aplicació són els següents:

- Permetre el registre i l'autenticació d'usuaris al sistema.
- Permetre que l'administrador del sistema tingui accés al manteniment dels usuaris i les partides de pòquer.
- Permetre als usuaris inscriure's a les partides de pòquer, i posteriorment, permetre l'accés a l'àrea de joc.
- Permetre l'accés a l'àrea de joc quan s'hagin inscrit 2 o més jugadors. Les partides són exclusivament multi jugador (concretament de 2 a 8 jugadors).
- L'aplicació haurà de controlar els casos en que un jugador abandoni voluntàriament o involuntàriament la partida per permetre que la resta de jugadors puguin continuar.
- Cada usuari registrat a una partida estarà sempre implicat en aquesta encara que no estigui connectat. La lògica del joc respectarà el torn de tots els jugadors registrats. No obstant, aquells que no estiguin connectats només consumiran les apostes mínimes obligatòries.

## 1.2. Objectius del projecte

L'objectiu del projecte es basa principalment en la realització d'una aplicació fent servir els coneixements adquirits al llarg de la carrera, emmarcats dins l'àmbit de les tecnologies oferides per la plataforma Microsoft .NET.

Més concretament i d'acord amb l'àrea escollida, l'objectiu funcional es centra en una aplicació web per a un casino virtual, que permeti el registre, l'autenticació i el manteniment d'usuaris, i ofereixi a aquests un àrea de jocs.

Com a objectius no funcionals, es planteja la utilització de la tecnologia ASP.NET, la plataforma Silverlight, i AJAX com a tècnica de desenvolupament web.

## 1.3. Planificació del projecte

### 1.3.1. Fites

#### **Pla de projecte**

**del 25/02/2010 al 10/03/2010**

- Escollir tema TFC
- Instal·lació programari
- Planificació del projecte
- Elaboració PAC1

#### **Anàlisi de requeriments**

**del 10/03/2010 al 18/03/2010**

- Anàlisi requeriments funcionals
- Anàlisi requeriments no funcionals
- Elaboració document

#### **Prototip**

**del 19/03/2010 al 26/03/2010**

- Elaboració prototip aplicació
- Elaboració document

#### **Disseny**

**del 27/03/2010 al 07/04/2010**

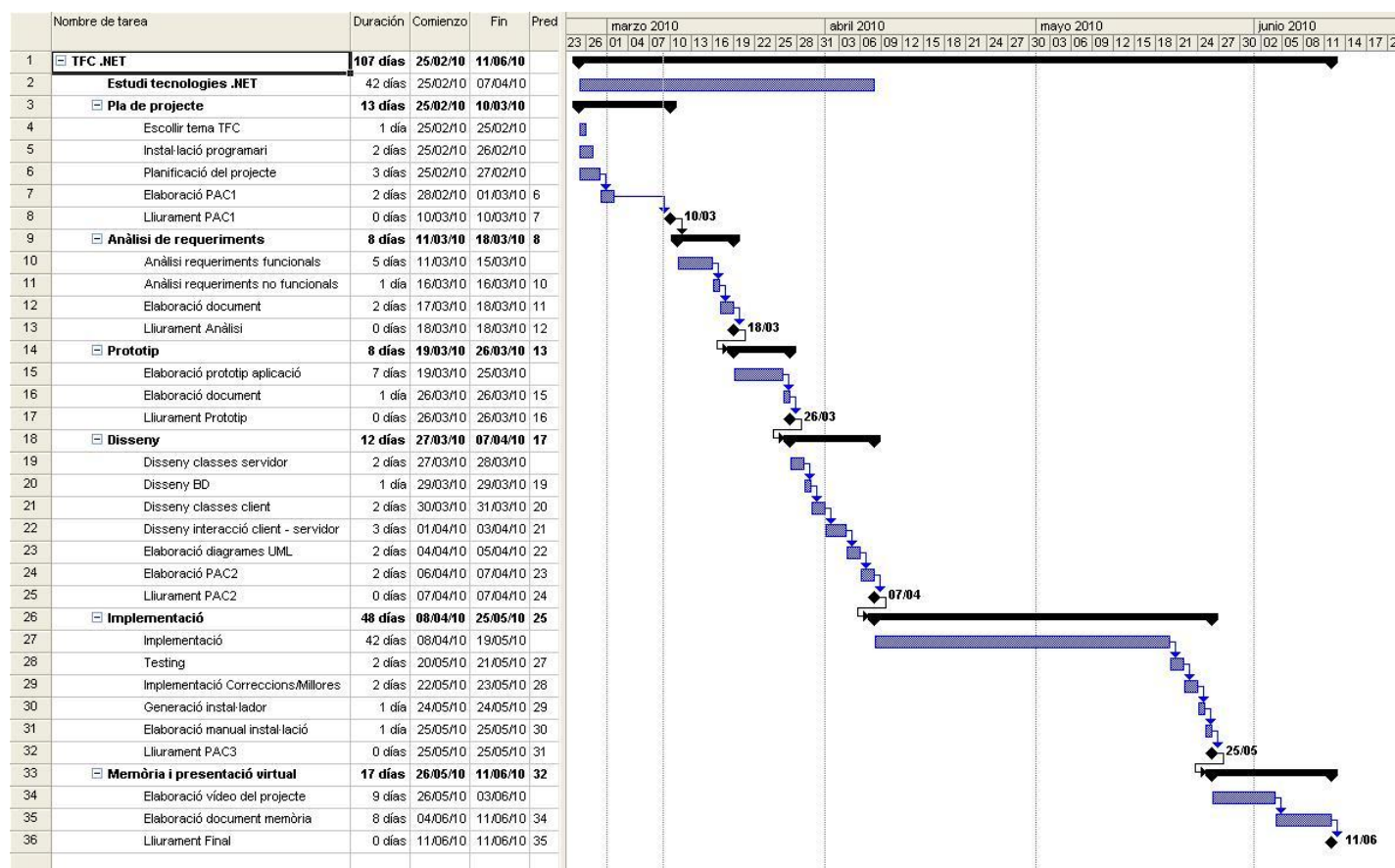
- Disseny classes servidor
- Disseny BD
- Disseny classes client
- Disseny interacció client - servidor
- Elaboració diagrames UML
- Elaboració PAC2

**Implementació****del 08/04/2010 al 25/05/2010**

- Implementació
- Testing
- Implementació Correccions/Millores
- Generació instal·lador
- Elaboració manual instal·lació

**Memòria i presentació virtual****del 26/05/2010 al 11/06/2010**

- Elaboració vídeo del projecte
- Elaboració document memòria
- Lliurament Final

**1.3.2. Diagrama de Gantt**

## 1.4. Productes obtinguts

Els productes obtinguts en el decurs del TFC són els següents:

### Pla de treball

Es tracta del document inicial que inclou la descripció del projecte, l'àmbit de desenvolupament, els requeriments funcionals bàsics, una breu introducció del joc a desenvolupar, i la planificació del projecte.

### Anàlisi i disseny

Un document que compren una descripció en detall de l'aplicació a desenvolupar, l'anàlisi de requeriments funcionals i tècnics, i també els casos d'ús.

La part corresponent al disseny inclou el disseny de l'arquitectura de l'aplicació, el disseny de les classes amb els corresponents diagrames, un prototip de la interfície gràfica, i el disseny de la base de dades.

### Implementació

Aquest producte consta de tot el codi font de la implementació, juntament amb un script de creació dels objectes de la base de dades, un manual d'instal·lació i un altre manual d'usuari.

### Memòria i presentació

Consta del present document, i un vídeo que mostra els trets principals del projecte.

## 2. Anàlisi

Per maximitzar la claredat de l'anàlisi funcional s'han dividit les funcionalitats entre un sistema principal i un subsistema de partida de joc que es detallen a continuació.

### 2.1. Anàlisi de requeriments funcionals

#### 2.1.1. Sistema principal

Aquest subsistema serà l'encarregat de gestionar:

- El registre d'usuaris.
- L'autenticació i la desconexió dels usuaris al sistema.
- La personalització dels menús i controls de les interfícies gràfiques d'usuari en funció del rol de l'usuari connectat.
- La gestió de les operacions CRUD de les diferents entitats.
- Les estadístiques de partides.
- El registre de participació d'usuaris a les partides.
- L'accés a les partides.

#### 2.1.2. Subsistema de partida

Aquest subsistema serà l'encarregat de gestionar el desenvolupament del joc així com les diferents interaccions entre els jugadors dins l'àmbit d'una partida.

## 2.2. Anàlisi de requeriments tècnics

L'aplicació PoquerSuau haurà d'estar enterament desenvolupada dins l'àmbit de la tecnologia .NET. Més concretament, i en favor d'una interfície gràfica d'usuari prou atractiva, les pantalles es desenvoluparan amb Silverlight i JavaScript, utilitzant aquest últim per tractament d'excepcions, missatges, etc.

Després d'analitzar els requeriments funcionals, cal fer esment de la relació de comunicació entre client i servidor. L'aplicació PoquerSuau ha de mantenir 2 canals de comunicació client-servidor:

- **Canal Usuari – Servidor**, que permetrà a un usuari interaccionar amb el sistema: autenticació, manteniments, consultes, accés a les partides, accions dins de l'àmbit d'una partida, etc.
- **Canal Rellotge – Servidor**, que dins l'àmbit del joc farà possible que un jugador, sense efectuar cap acció a la interfície gràfica d'usuari, pugui percebre les accions executades per la resta de jugadors i l'estat actual del joc: repartiment de cartes, canvi de torn, última aposta efectuada, ... Aquesta serà una comunicació molt freqüent que permetrà mantenir "viu" el joc, traslladant al jugador la sensació d'interacció a temps real amb la resta de jugadors.

És per tant prou obvi que s'hauran d'utilitzar serveis web per poder garantir aquests canals de comunicació. No obstant, aquests aspectes s'acabaran de concretar a la fase de disseny.



## 2.3. Requeriments tècnics de maquinari

- Un servidor amb el rol d'IIS que pugui oferir servei a una quantitat important d'usuaris.
- Un servidor amb el rol de sistema gestor de base de dades que haurà de donar servei a l'anterior.
- Ordinadors client amb sistema operatiu Windows XP, Windows Vista, Windows 7, Mac, Linux, que puguin albergar un navegador compatible amb Silverlight: Internet Explorer, Firefox, Safari, Opera, Chrome i evidentment estiguin connectats a Internet.

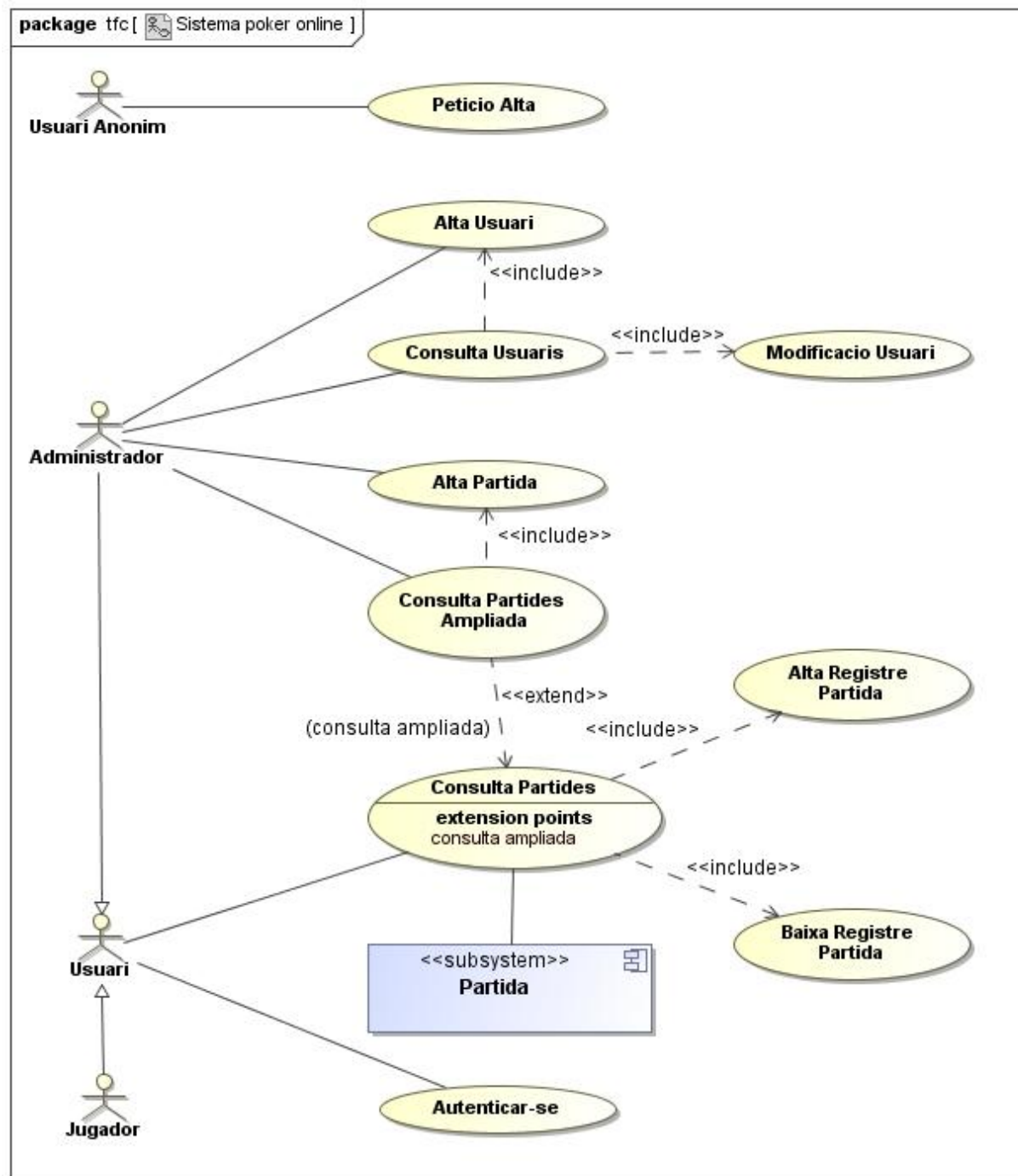
## 2.4. Casos d'ús

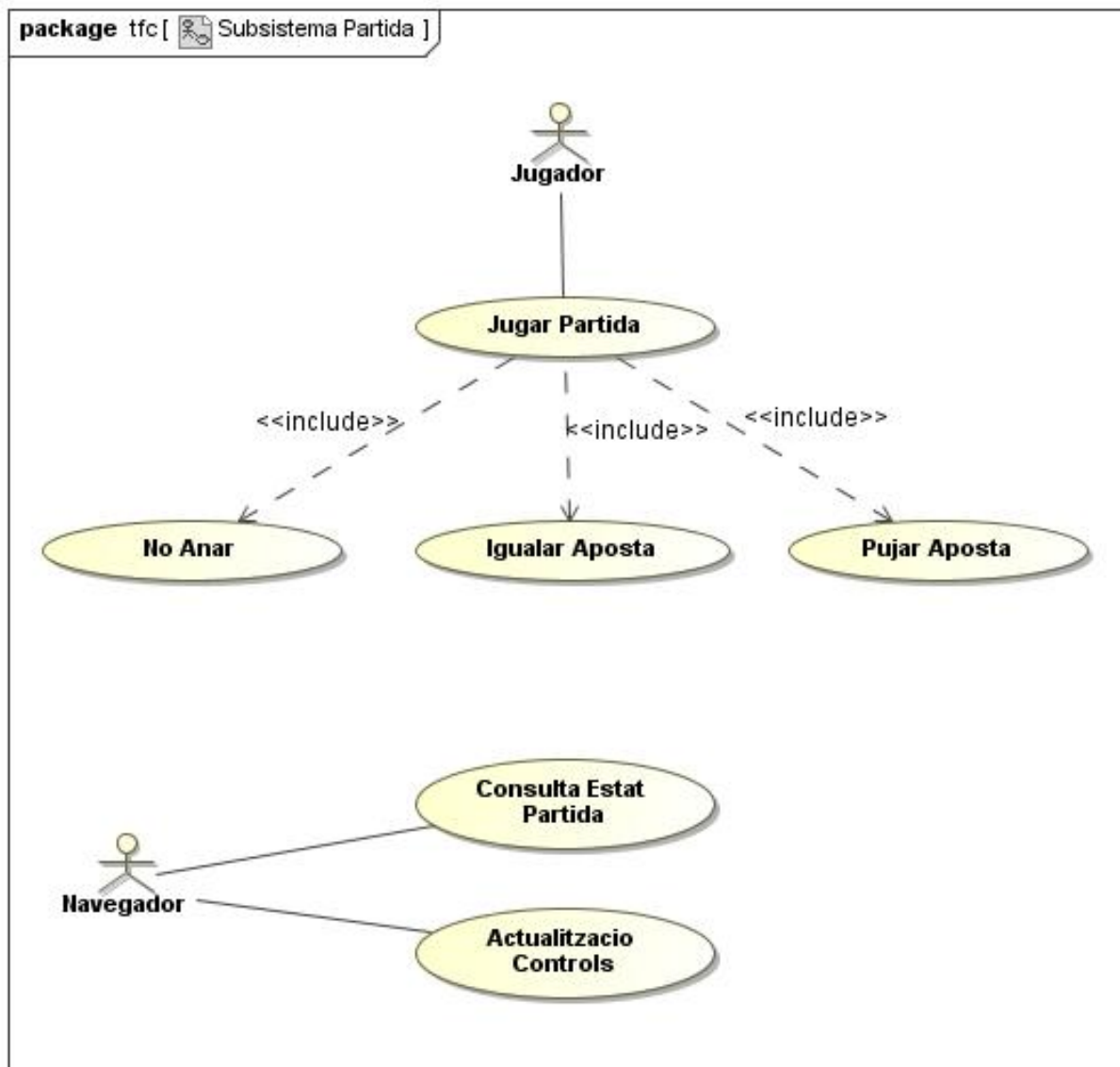
### 2.4.1. Actors

Els actors que intervenen als casos d'ús són els següents:

- **Usuari anònim:** és un usuari que accedeix a l'aplicació web però no està autenticat al sistema.
- **Usuari:** és un usuari autenticat al sistema.
- **Administrador:** és un usuari autenticat al sistema que, a més, té assignat el perfil d'administrador del sistema.
- **Jugador:** és un **usuari** dins l'àmbit d'una partida concreta.
- **Rel·lotge:** és el propi navegador per aquells casos d'ús en que periòdicament (cada 2 segons, per exemple) va comprovant l'estat de joc, recollint els diferents esdeveniments produïts per altres jugadors, i permetent que s'actualitzin els controls corresponents en conseqüència.

## 2.4.2. Diagrames de casos d'ús





## 2.4.2. Descripció dels casos d'ús

Cas d'ús	Peticio Alta
Descripció	Permet a un usuari no autenticat realitzar una petició d'alta al sistema mitjançant un formulari de registre.
Actors	Usuari Anonim
Pre-condicions	L'actor no està autenticat al sistema.
Post-condicions	S'enregistra la petició d'alta d'usuari al sistema.
Procés principal	Pantalla inicial.
Casos d'ús relacionats	

Cas d'ús	Autenticar-se
Descripció	Permet a un usuari accedir al sistema mitjançant la introducció del nom d'usuari i la contrasenya. En funció del rol de l'usuari autenticat (usuari o administrador) s'activaran les opcions de menú corresponents.
Actors	Usuari, Administrador
Pre-condicions	L'actor no està autenticat al sistema.
Post-condicions	L'actor està autenticat al sistema, o les dades introduïdes no són vàlides.
Procés principal	Pantalla inicial.
Casos d'ús relacionats	

Cas d'ús	Consulta Usuaris
Descripció	Mostra una grid amb les dades dels usuaris existents al sistema. Permet filtrar les dades segons el nom o l'estat de l'usuari.
Actors	Administrador
Pre-condicions	L'actor esta autenticat al sistema.
Post-condicions	Es mostren els usuaris del sistema.
Procés principal	Menú d'opcions principal
Casos d'ús relacionats	Alta Usuari, Modificacio Usuari, Baixa Usuari

Cas d'ús	Alta Usuari
Descripció	<p>Permet a l'actor donar d'alta un nou usuari al sistema. Aquesta operació pot obeir a una alta esporàdica, o pot ser la resposta d'una petició d'alta d'usuari.</p> <p>En el moment de validar l'alta d'usuari, aquest automàticament obtindrà un crèdit inicial (fictici) que li permetrà participar a les partides.</p>
Actors	Administrador
Pre-condicions	L'actor esta autenticat al sistema.
Post-condicions	Cap
Procés principal	Menú d'opcions principal, cas d'ús Consulta Usuaris
Casos d'ús relacionats	Consulta Usuaris

Cas d'ús	Modificació Usuari
Descripció	Permet a l'actor modificar les dades d'un usuari. Permet també donar de baixa un usuari - atès que les baixes són lògiques - mitjançant la modificació de l'Estat de l'usuari (BAIXA).
Actors	Administrador
Pre-condicions	L'actor esta autenticat al sistema i ha seleccionat un usuari vàlid.
Post-condicions	Els canvis efectuats per l'actor es fan persistents o alguna de les dades introduïdes no és vàlida.
Procés principal	Cas d'ús Consulta Usuaris
Casos d'ús relacionats	Consulta Usuaris

Cas d'ús	Consulta Partides
Descripció	<p>Permet a l'actor visualitzar les partides pendents d'iniciar, i a les quals l'usuari pot participar. També mostra les partides on s'ha registrat l'usuari prèviament.</p> <p>També mostra, a mode d'estadística, les partides on ha participat l'usuari.</p>
Actors	Usuari, Administrador
Pre-condicions	L'actor esta autenticat al sistema.
Post-condicions	Es mostren les partides sense començar.
Procés principal	Menú d'opcions principal
Casos d'ús relacionats	Alta Registre Partida, Baixa Registre Partida

Cas d'ús	Alta Registre Partida
Descripció	Permet a un usuari apuntar-se a una partida per poder participar-hi. Evidentment només podrà participar a les partides pendents de començar i on hi hagin places disponibles.
Actors	Usuari
Pre-condicions	L'actor ha seleccionat una partida no iniciada amb places lliures, i té crèdit suficient per pagar el preu de la partida.
Post-condicions	L'actor s'enregistra al sistema com a jugador de la partida amb un número de plaça concret.
Procés principal	cas d'ús Consulta Partides
Casos d'ús relacionats	Consulta Partides, Actualització SaldoUsuari

Cas d'ús	Baixa Registre Partida
Descripció	Permet a un usuari desregistrar-se d'una partida no iniciada on s'havia registrat prèviament. Com a conseqüència, es fa un reemborsament al crèdit de l'usuari corresponent al preu de la partida.
Actors	Usuari
Pre-condicions	L'actor ha seleccionat una partida on estava prèviament registrat.
Post-condicions	L'usuari queda desregidrat de la partida, i el preu d'aquesta s'afegeix al seu crèdit.
Procés principal	Cas d'ús Consulta Partides
Casos d'ús relacionats	Consulta Partides, Actualització SaldoUsuari

Cas d'ús	Consulta Partides Ampliada
Descripció	A més de les funcionalitats oferides pel cas d'ús Consulta Partides, permet a l'actor visualitzar totes les partides incloses al sistema filtrant per usuari, data, estat, etc.
Actors	Administrador
Pre-condicions	L'actor està autenticat al sistema.
Post-condicions	Es mostren les partides d'acord amb els filtres seleccionats per l'actor.
Procés principal	Menú d'opcions principal
Casos d'ús relacionats	Alta Partida

Cas d'ús	Alta Partida
Descripció	Permet a l'actor donar d'alta una nova partida de pòquer.
Actors	Administrador
Pre-condicions	L'actor esta autenticat al sistema.
Post-condicions	Es crea una nova partida al sistema o les dades introduïdes no són vàlides.
Procés principal	Menú d'opcions principal, cas d'ús Consulta Partides Ampliada
Casos d'ús relacionats	Consulta Partides Ampliada

Cas d'ús	Jugar Partida
Descripció	Permet a l'actor accedir a l'escenari d'una partida de pòquer a la qual està registrat, a partir de l'hora de començament d'aquesta.
Actors	Usuari
Pre-condicions	L'usuari està autenticat al sistema, està registrat a la partida i la partida està iniciada.
Post-condicions	L'usuari accedeix a l'escenari de la partida i pren el rol de jugador.
Procés principal	Cas d'ús Consulta Partides
Casos d'ús relacionats	Consulta Partides, No Anar,

Cas d'ús	No Anar
Descripció	<p>Dins l'escenari d'una partida i dins del torn corresponent com a jugador, permet a aquest abandonar la ronda actual.</p> <p>Quan el jugador executa aquesta acció, el navegador actualitza els controls de l'escenari de joc adequadament, i envia un senyal al servidor per actualitzar l'estat de la partida, de manera que la resta de navegadors (d'altres jugadors) puguin també actualitzar els controls dins l'escenari de joc corresponent.</p>
Actors	Jugador
Pre-condicions	L'opció "No Anar" està disponible a l'escenari de joc.
Post-condicions	El jugador perd les fitxes invertides i la possibilitat d'igualar o pujar apostes dins l'àmbit de la ronda de joc actual. El torn passa al següent jugador.
Procés principal	Jugar Partida
Casos d'ús relacionats	Jugar Partida, Igualar Aposta, Pujar Aposta



Cas d'ús	Igualar Apostes
Descripció	<p>Dins l'escenari d'una partida i dins del torn corresponent com a jugador, permet a aquest igualar les apostes aplicades per altres jugadors, de manera que continua essent candidat a guanyar la ronda actual.</p> <p>Quan el jugador executa aquesta acció, el navegador actualitza els controls de l'escenari de joc adequadament, i envia un senyal al servidor per actualitzar l'estat de la partida, de manera que la resta de navegadors (d'altres jugadors) puguin també actualitzar els controls dins l'escenari de joc corresponent.</p>
Actors	Jugador
Pre-condicions	L'opció "Igualar Aposta" està disponible a l'escenari de joc.
Post-condicions	El jugador es manté a la ronda actual i passa el torn al següent jugador.
Procés principal	Jugar Partida
Casos d'ús relacionats	Jugar Partida, No Anar, Pujar Aposta

Cas d'ús	Pujar Aposta
Descripció	<p>Dins l'escenari d'una partida i dins del torn corresponent com a jugador, permet a aquest pressionar a la resta de jugadors de manera que hagin d'igualar la seva aposta si volen continuar actius a la ronda actual.</p> <p>Quan el jugador executa aquesta acció, el navegador actualitza els controls de l'escenari de joc adequadament, i envia un senyal al servidor per actualitzar l'estat de la partida, de manera que la resta de navegadors (d'altres jugadors) puguin també actualitzar els controls dins l'escenari de joc corresponent.</p>
Actors	Jugador
Pre-condicions	L'opció "Pujar Aposta" està disponible a l'escenari de joc.
Post-condicions	El jugador es manté a la ronda actual i passa el torn al següent jugador.
Procés principal	Jugar Partida
Casos d'ús relacionats	Jugar Partida, No Anar, Igualar Aposta

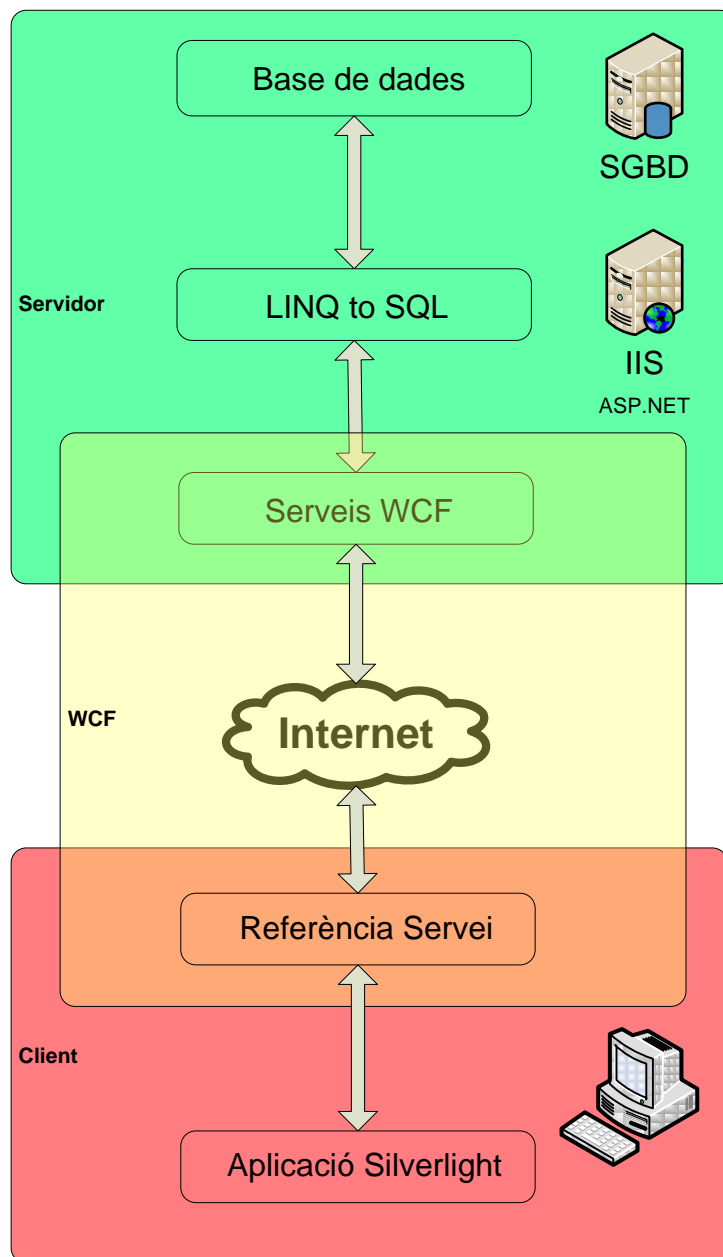
Cas d'ús	Consulta Estat Partida
Descripció	Dins l'escenari de joc el navegador periòdicament va enviant peticions de consulta al servidor obtenint com a resultat l'estat del joc actual, d'acord amb les accions realitzades pels jugadors.
Actors	Navegador
Pre-condicions	Existeix un jugador autenticat al sistema mitjançant el navegador.
Post-condicions	El navegador (l'aplicació client que s'executa al navegador) rep l'estat de la partida.
Procés principal	
Casos d'ús relacionats	Actualitzar Controls

Cas d'ús	Actualitzar Controls
Descripció	Dins l'escenari de joc el navegador va actualitzant els controls d'acord amb l'estat actual de la partida i les accions realitzades pel jugador durant el seu torn.
Actors	Navegador
Pre-condicions	Cap
Post-condicions	L'escenari s'actualitza en coherència amb l'estat de la partida i les accions realitzades pel jugador durant el seu torn.
Procés principal	
Casos d'ús relacionats	Consulta Estat Partida

## 3. Disseny Tècnic

### 3.1. Disseny de l'arquitectura de l'aplicació

S'ha projectat PoquerSuau com una aplicació orientada a serveis, de manera que totes les comunicacions entre client i servidor es produiran a través de la tecnologia WCF.



Els usuaris accediran a l'aplicació a través d'un navegador web.

Des de l'autenticació al sistema fins a la gestió del joc, les accions de l'usuari seran captades per l'aplicació Silverlight, on tot el "code behind" s'implementarà amb el llenguatge C#.

Tal i com s'ha comentat a l'anàlisi, s'utilitzaran mètodes Javascript per implementar algunes funcionalitats relacionades amb la interfície gràfica d'usuari.

Silverlight interaccionarà amb el servidor mitjançant referències de serveis WCF. Aquestes referències de serveis són les interfícies de client encarregades d'establir la comunicació amb les interfícies de servidor que ofereixen els serveis WCF.

Cada servei WCF ofert pel servidor web es comunicarà amb el servidor de base de dades mitjançant instàncies de classes LINQ to SQL.

Pel que fa a l'escenari del joc, cal esmentar que tots els esdeveniments produïts pels diferents actors – els jugadors – s'hauran de fer persistents molt freqüentment en el decurs de cada partida, de manera que tots els usuaris (jugadors) puguin veure reflectits aquests esdeveniments, al seu respectiu navegador. Aquesta actualització contínua dels events produïts es farà possible gràcies a l'actor rellotge (navegador) que, mitjançant els serveis wcf, anirà recollint molt freqüentment l'estat actual de la partida i actualitzant l'escenari en conseqüència.

## 3.2. Disseny de les classes

Seguint les línies de l'arquitectura de l'aplicació, les classes es dividiran en quatre paquets:

- **Classes de la interfície gràfica d'usuari**

Són les classes de Silverlight que inclouen tota la interfície gràfica de l'aplicació. Aquestes classes utilitzaran instàncies de classes d'entitats així com referències a serveis de comunicació.

- **Classes de comunicació**

Són les classes que ofereixen els serveis WCF. Aquests serveis permeten la comunicació entre els objectes de la interfície d'usuari i els objectes del model de base de dades.

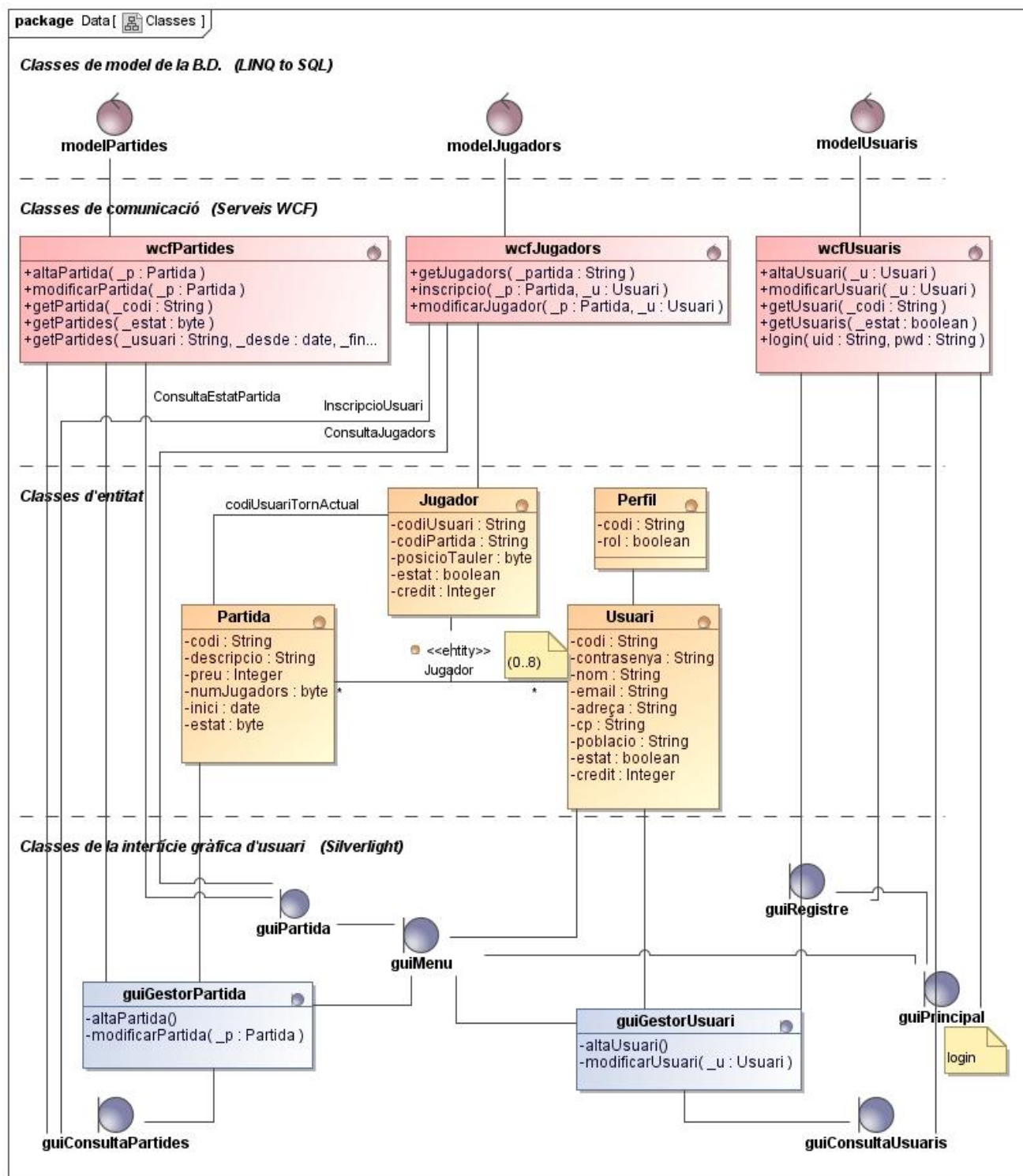
- **Classes de model de la base de dades**

Són les classes LINQ to SQL, que tindran mapejats els objectes persistents.

- **Classes d'entitat**

Són les classes que representen entitats de la lògica de negoci. Les instàncies d'aquestes classes són transportades de la banda del client a la banda del servidor i a l'inrevés a través de les classes de comunicació.

### 3.3. Diagrama de classes



## 3.4. Fitxes CRC de les classes

A continuació es detallen les classes d'entitat i les classes de comunicació en fitxes CRC (classe – responsabilitats – col·laboracions)

### 3.4.1. Classes d'entitat

Classe	Partida
Descripció	Representa una partida de pòquer de l'aplicació PoquerSuau
Responsabilitats	Identificar unívocament una partida
Col·laboracions	Jugador
Atributs	<ul style="list-style-type: none"> <li>- codi: string</li> <li>- descripcio: string</li> <li>- preu: integer</li> <li>- numJugadors: byte</li> <li>- inici: date</li> <li>- estat: byte //Pendent/Iniciada/Finalitzada/Anul·lada</li> <li>- codiUsuariTornActual: string //Permet conèixer el jugador que té el torn en un moment donat</li> </ul>
Mètodes	+ getters i setters de tots els atributs privats

Classe	Usuari
Descripció	Representa un usuari, administrador o no, actiu o no de l'aplicació PoquerSuau
Responsabilitats	Identificar unívocament un usuari del sistema
Col·laboracions	Jugador
Atributs	<ul style="list-style-type: none"> <li>- codi: string</li> <li>- contrasenya: string</li> <li>- nom: string</li> <li>- email: string</li> <li>- adreça: string</li> <li>- cp: string</li> <li>- poblacio: string</li> <li>- estat: byte //Actiu/Inactiu/Bloquejat</li> <li>- credit: int //crèdit disponible per inscriure's a les partides</li> <li>- administrador: boolean</li> </ul>
Mètodes	+ getters i setters de tots els atributs privats

Classe	Jugador
Descripció	Representa un usuari del sistema dins l'àmbit d'una partida on està inscrit
Responsabilitats	Identificar unívocament un jugador d'una partida de pòquer
Col·laboracions	Partida, Usuari
Atributs	<ul style="list-style-type: none"> <li>- codiPartida: string</li> <li>- codiUsuari: string</li> <li>- posicioTauler: byte</li> <li>- estat: int //Absent/AbandonantRonda/Actiu/Eliminat</li> <li>- credit: int //crèdit disponible a la partida</li> </ul>
Mètodes	+ getters i setters de tots els atributs privats

### 3.4.2. Classes de comunicació

Classe	wcfPartides
Descripció	És un servei wcf que permet establir i recuperar dades persistents relacionades amb una o vàries partides de pòquer del sistema
Responsabilitats	Enllaçar amb el model de base de dades modelPartides permetent la lectura i escriptura de dades persistents, així com permetre el flux de dades cap a les classes d'interfície gràfica de partida
Col·laboracions	modelPartides, guiPartida, guiGestorPartida, guiConsultaPartides
Mètodes	<ul style="list-style-type: none"> <li>+ altaPartida(_p: Partida)</li> <li>+ modificarPartida(_p: Partida)</li> <li>+ getPartida(_codi: string)</li> <li>+ getPartides(_estat: byte)</li> <li>+ getPartides(_u: Usuari, _desde: date, _fins: date)</li> </ul>



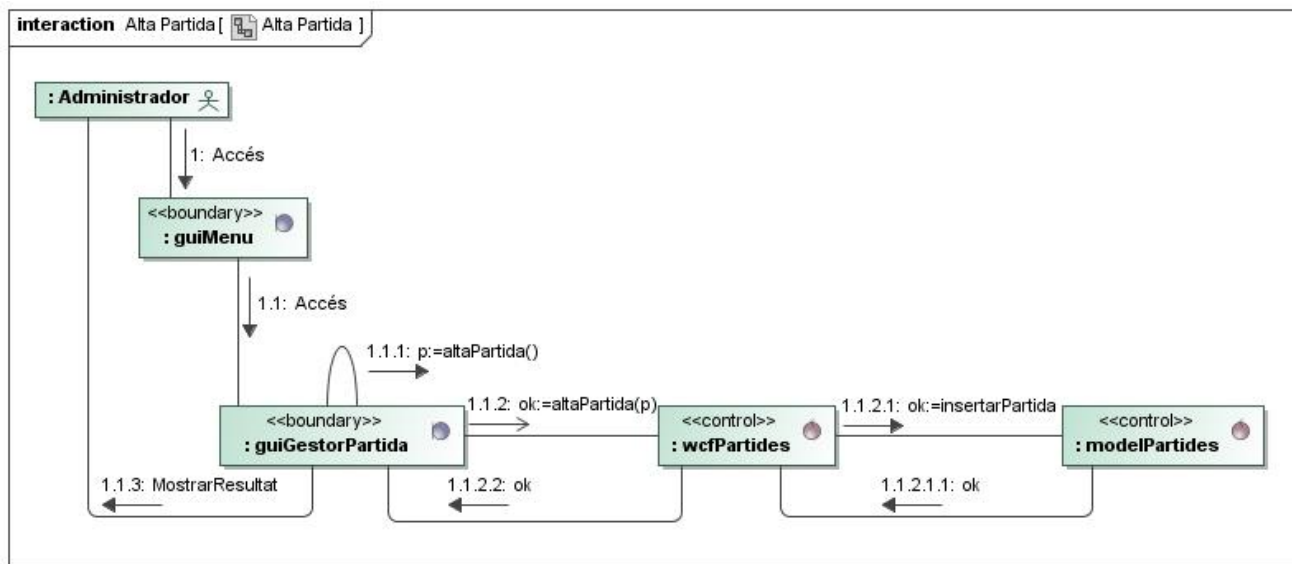
Classe	wcfUsuaris
Descripció	És un servei wcf que permet establir i recuperar dades persistents relacionades amb un o varis usuaris del sistema
Responsabilitats	Enllaçar amb el model de base de dades modelUsuaris permetent la lectura i escriptura de dades persistents, així com permetre el flux de dades cap a les classes d'interfície gràfica d'usuari
Col·laboracions	modelUsuaris, guiRegistre, guiPrincipal, guiGestorUsuari, guiConsultaUsuaris
Mètodes	+ altaUsuari(_u: Usuari) + modificarPartida(_u: Usuari) + getUsuari(_codi: string) + getJugadors(_estat: byte) + login(uid: string, pwd: string)

Classe	wcfJugadors
Descripció	És un servei wcf que permet establir i recuperar dades persistents relacionades amb un o varis jugadors d'una partida
Responsabilitats	Enllaçar amb el model de base de dades modelJugadors permetent la lectura i escriptura de dades persistents, així com permetre el flux de dades cap a les classes d'interfície gràfica guiPartida i guiGestorPartides
Col·laboracions	modelJugadors, guiPartida, guiGestorPartides
Mètodes	+ getJugadors(_p: Partida) + inscripcio(_p: Partida, _u: Usuari) + modificarJugador(_j: Jugador)

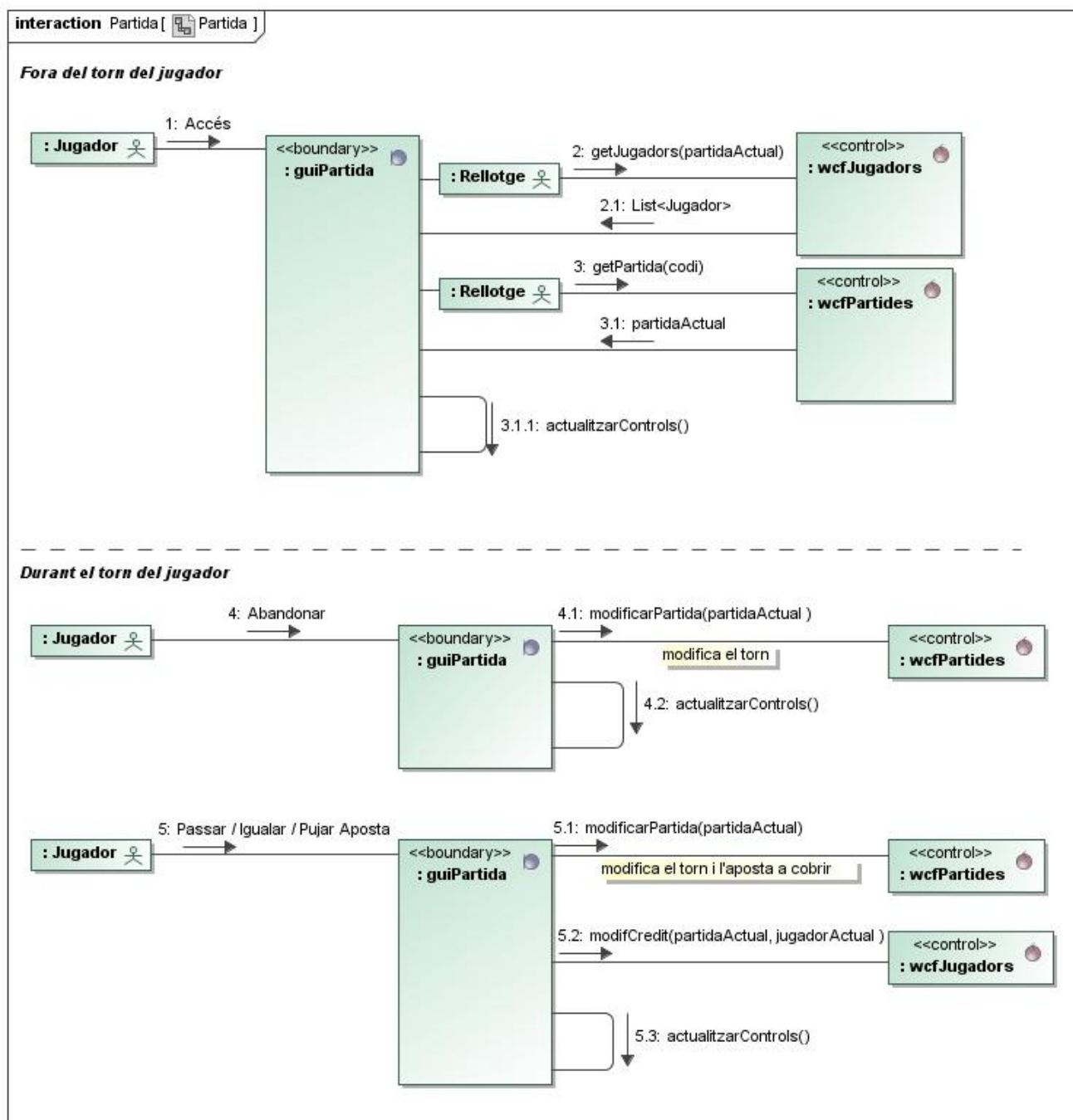
## 3.5. Diagrames de col·laboració

A continuació s'exposen alguns diagrames de col·laboració

### 3.5.1. Diagrama de col·laboració del cas d'ús Alta Partida



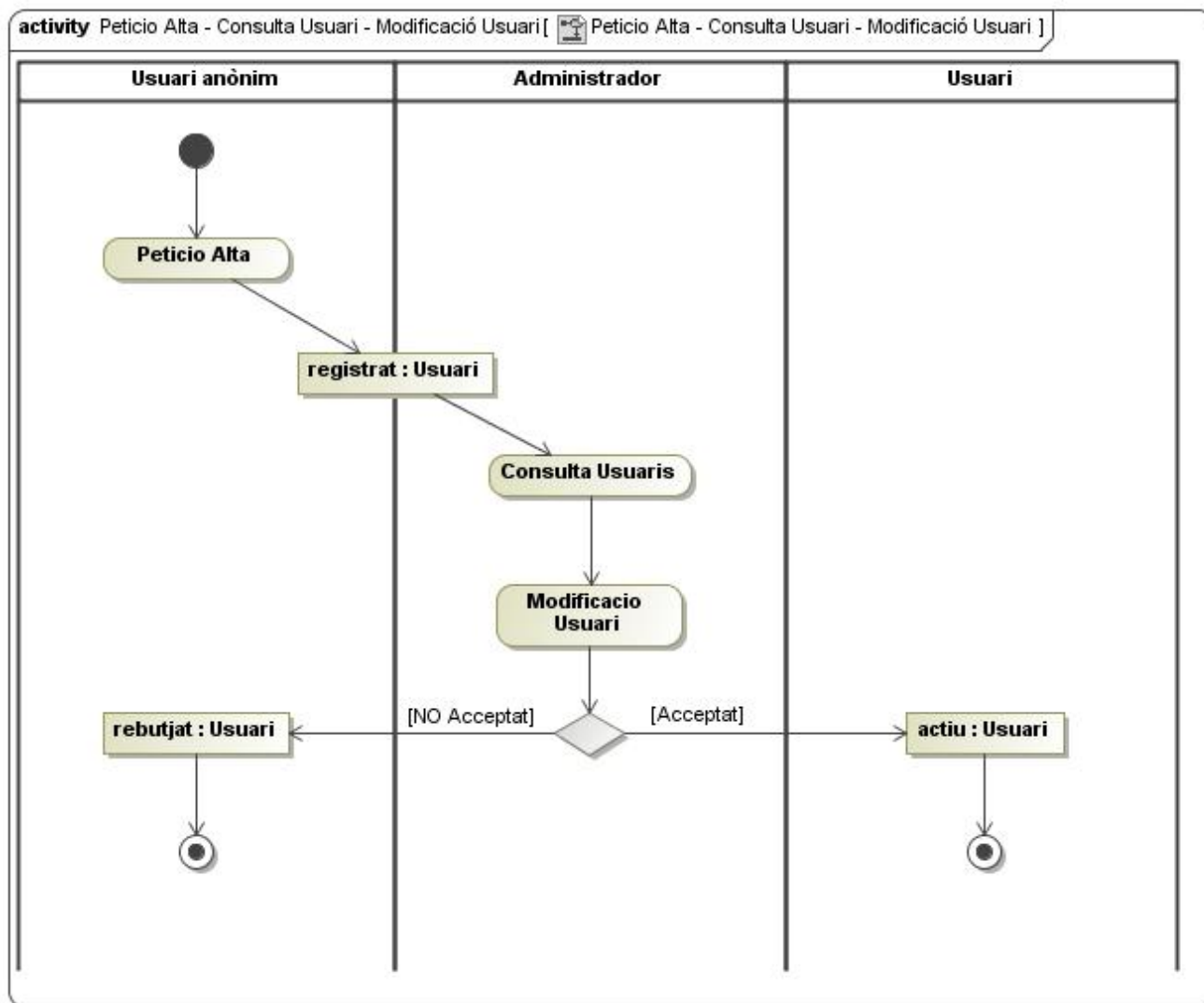
### 3.5.2. Diagrama de col·laboració del subsistema Partida



## 3.6. Diagrames d'activitats

Amb l'objectiu d'aclarir el funcionament d'alta d'usuaris, a continuació es mostra un diagrama d'activitats relacionat.

### 3.6.1. Diagrama d'activitats relacionades amb l'alta d'un usuari

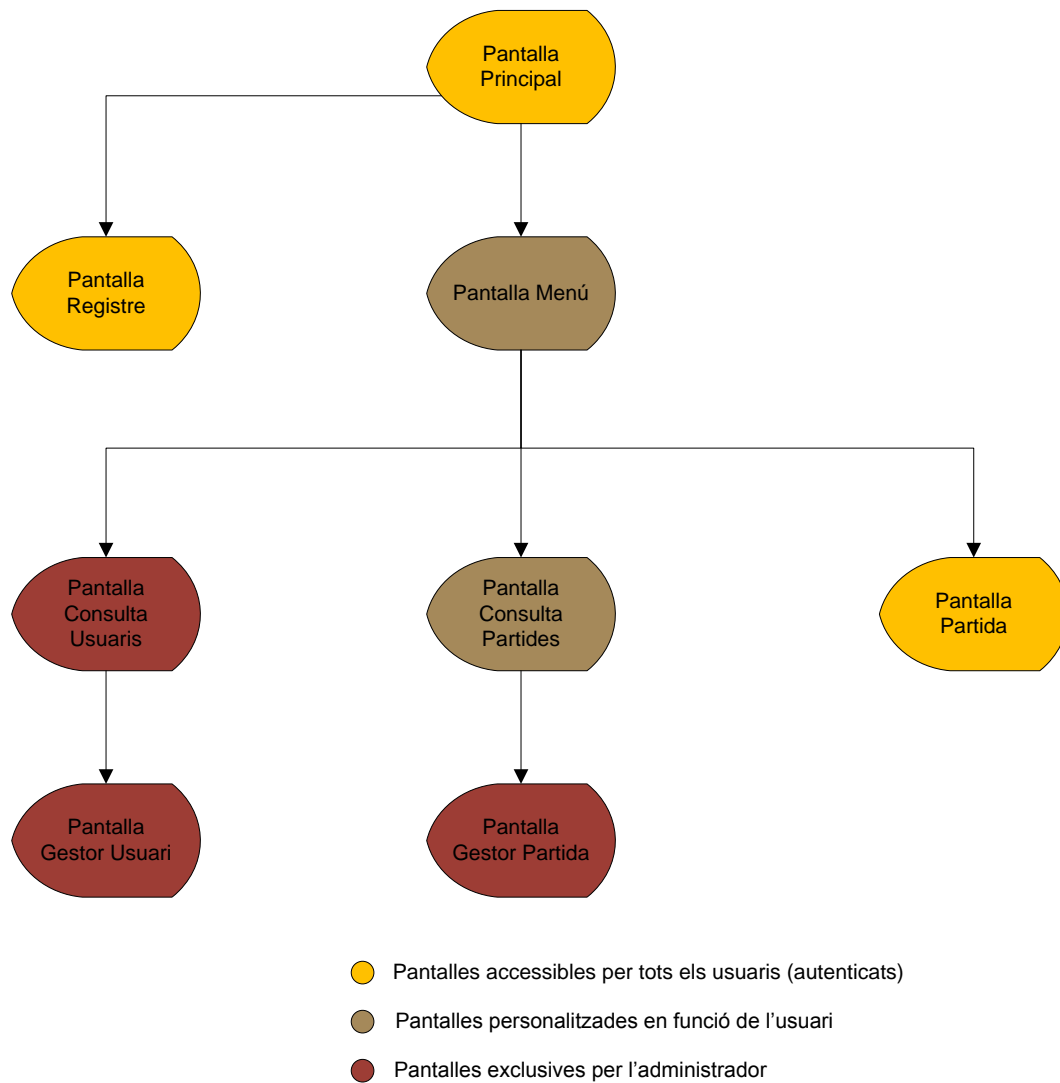


Com es pot interpretar a partir del diagrama, un usuari no és actiu al sistema fins que no el valida l'administrador.

A banda, cal aclarir també que un usuari no podrà accedir al menú principal – i per tant no podrà participar de cap partida – fins que no estigui actiu.

## 3.7. Disseny de la interfície gràfica d'usuari

### 3.7.1. Diagrama de flux de les pantalles



## 3.8. Disseny de la base de dades

El model escollit per la implementació de la capa de persistència de l'aplicació és el model relacional de base de dades. Concretament el sistema gestor serà SQL Server 2008.

### 3.8.1. Descripció de les taules

Atès que el nombre de perfils possibles es redueixen a 2 – perfil usuari i perfil administrador – s'opta per unificar tots els usuaris dins una mateixa taula que comptarà amb l'atribut **administrador** de tipus booleà, que distingirà el rol de l'usuari.

Tanmateix, pel que fa als usuaris registrats que no s'han activat al sistema, s'afegeix un altre camp amb nom **estat** que distingirà els usuaris actius, inactius i bloquejats.

El disseny de taules resultants és el següent:

**Usuari** (codi, contrasenya, nom, email, adreça, cp, poblacio, estat, credit)

**Partida** (codi, descripcio, preu, numJugadors, inici, estat, codiUsuariTornActual)

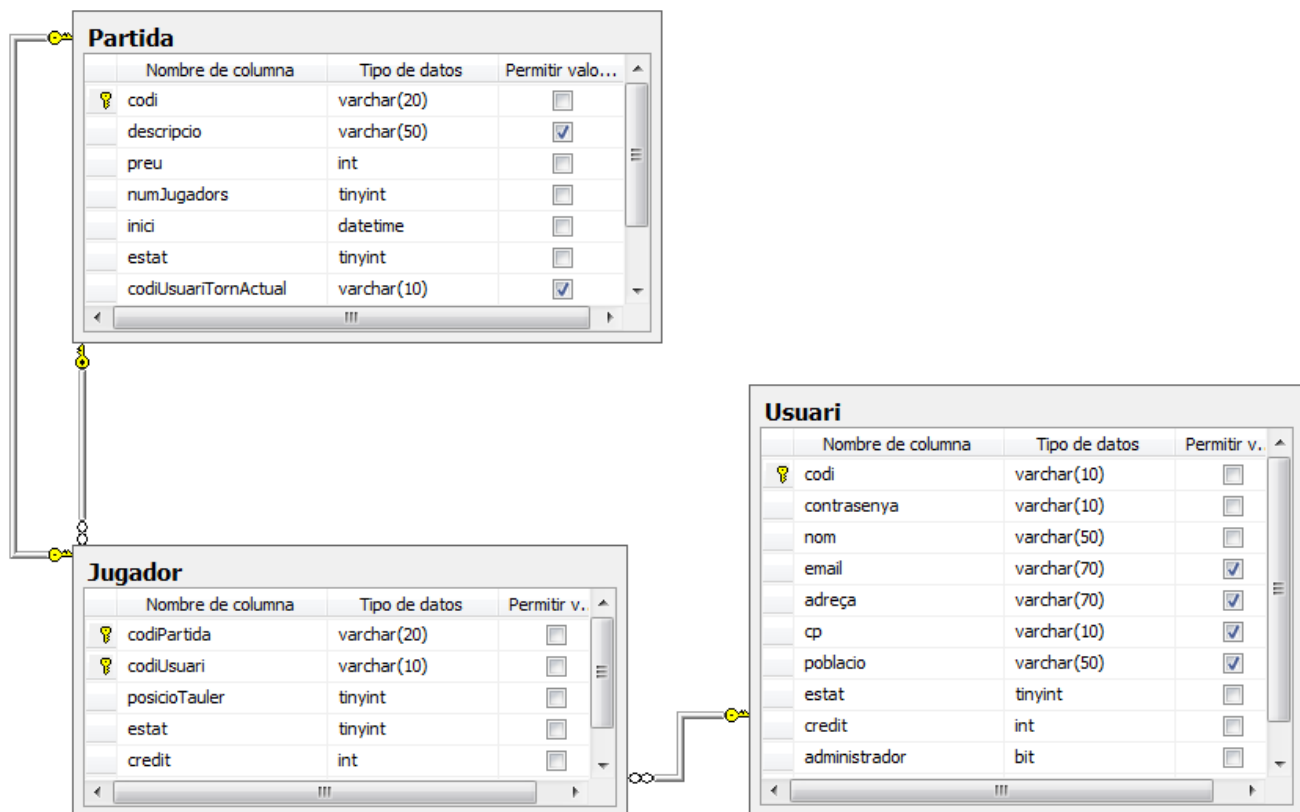
{codi, codiUsuariTornActual} és clau forana a Jugador. Es crea aquesta clau forana per garantir que l'usuari assignat com a jugador que té el torn sigui un dels jugadors de la pròpia partida.

**Jugador** (codiPartida, codiUsuari, posicioTauler, estat, credit)

{codiPartida} és clau forana a Partida

{codiUsuari} és clau forana a Usuari

### 3.7.2. Diagrama de la base de dades



## 4. Implementació

### 4.1. Modificacions respecte al disseny

Durant la fase d'implementació, algunes classes d'entitat i la base de dades han sofert algunes variacions respecte el plantejament realitzat al disseny tècnic.

#### 4.1.1. Variacions a les classes d'entitat

S'han incorporat nous atributs a les taules Partida i Jugador. Aquests atributs han permès minimitzar la complexitat d'alguns algorismes involucrats a la lògica del joc.

A continuació es mostren novament les fitxes CRC d'aquestes classes:

Classe	Partida
Descripció	Representa una partida de pòquer de l'aplicació PoquerSuau
Responsabilitats	Identificar unívocament una partida
Col·laboracions	Jugador
Atributs	<ul style="list-style-type: none"> <li>- codi: string</li> <li>- descripcio: string</li> <li>- preu: integer</li> <li>- numJugadors: byte</li> <li>- inici: date</li> <li>- estat: byte //Pendent/Iniciada/Finalitzada/Anul·lada</li> <li>- codiUsuariTornActual: string //Permet conèixer el jugador que té el torn en un moment donat</li> <li>- codiUsuariBoto: string //Conté el repartidor de la mà</li> <li>- codiUsuariCegaPetita: string //El jugador que paga la 1a. aposta mínima obligatòria</li> <li>- codiUsuariCegaGran: string //El jugador que paga la 2a. aposta mínima obligatòria</li> <li>- codiUsuariTestimoni: string //Permet mantenir la sincronització per a una partida</li> <li>- flop1: string //1a. carta comú</li> <li>- flop2: string //2a. carta comú</li> <li>- flop3: string //3a. carta comú</li> <li>- turn: string //4a. carta comú</li> <li>- river: string //5a. carta comú</li> <li>- pot: integer</li> <li>- apostaACobrir: integer</li> <li>- cegaPetita: int //import cega petita (1a. aposta mínima)</li> <li>- cegaGran: int //import cega petita (1a. aposta mínima)</li> <li>- horaRepartiment: datetime</li> <li>- horaFlop: datetime</li> <li>- horaTurn: datetime</li> <li>- horaRiver: datetime</li> <li>- horaFiRonda: datetime</li> </ul>
Mètodes	+ getters i setters de tots els atributs privats



Classe	Jugador
Descripció	Representa un usuari del sistema dins l'àmbit d'una partida on està inscrit
Responsabilitats	Identificar unívocament un jugador d'una partida de pòquer
Col·laboracions	Partida, Usuari
Atributs	<ul style="list-style-type: none"> <li>- codiPartida: string</li> <li>- codiUsuari: string</li> <li>- posicioTauler: byte</li> <li>- estat: int //Absent/AbandonantRonda/Actiu/Eliminat</li> <li>- credit: int //crèdit disponible a la partida</li> <li>- carta1: string</li> <li>- carta2: string</li> <li>- aposta: int</li> <li>- jugada: byte //Identifica la jugada aconseguida (parella, trio, ...)</li> <li>- combinacio: string //millor combinació de cartes aconseguida</li> <li>- c1: string //1a. Carta millor combinació</li> <li>- c2: string //2a. Carta millor combinació</li> <li>- c3: string //3a. Carta millor combinació</li> <li>- c4: string //4a. Carta millor combinació</li> <li>- c5: string //5a. Carta millor combinació</li> <li>- apostatRonda: int</li> <li>- posicioFinal: byte</li> </ul>
Mètodes	+ getters i setters de tots els atributs privats

#### 4.1.2. Variacions a les classes de comunicació i model de dades

Per fer més senzill el desenvolupament, les classes de model de la base de dades (LinQ to SQL) exposades al disseny s'han reunit en una única classe anomenada **model**. Anàlogament, tots els mètodes inclosos a les classes de comunicació wcfPartides, wcfUsuaris i wcfJugadors també s'han unificat en una única classe anomenada **wcf**.

#### 4.1.3. Variacions a la base de dades

Lògicament, a arrel dels nous atributs a les classes Partida i Jugador, les taules corresponents han estat modificades incorporant finalment nous camps. A més, s'ha afegit un camp tstamp de tipus timestamp a les taules Partida, Usuari, Jugador per permetre efectuar la modificació asíncrona per part del servei wcf.

A banda d'aquests camps, també s'han afegit dues noves taules, que han permès donar més protagonisme a l'SQL Server per resoldre algorismes relacionats amb la lògica de negoci. A continuació es defineixen aquestes taules:

##### **Baralla** (carta, ordre)

Aquesta taula conté sempre les 52 cartes que formen part del joc, és a dir, que el seu contingut és sempre el mateix.

## Partida\_Cartes (codiPartida, carta, ordre)

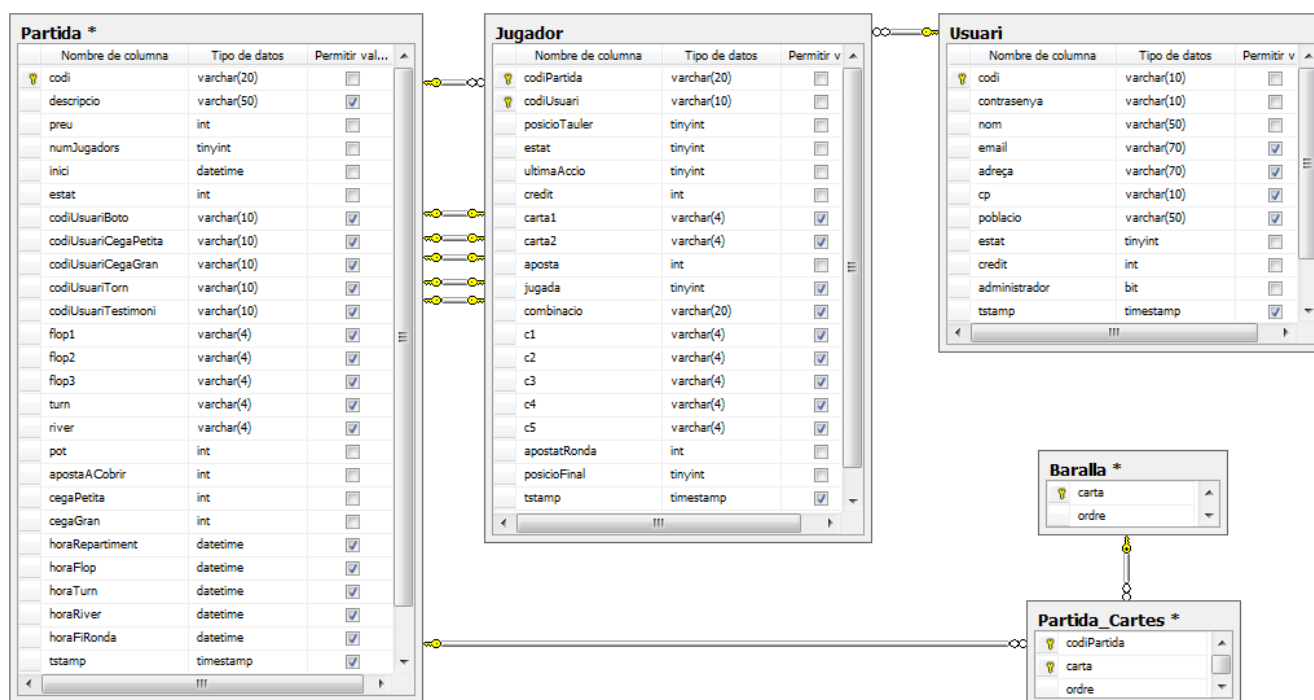
{codiPartida} és clau forana a Partida

{carta} és clau forana a Baralla

Al començament d'una partida, s'incorporen totes les cartes de la taula Baralla dins la taula Partida\_Cartes.

Un cop finalitzada una partida, els registres corresponents d'aquesta nova taula són totalment prescindibles.

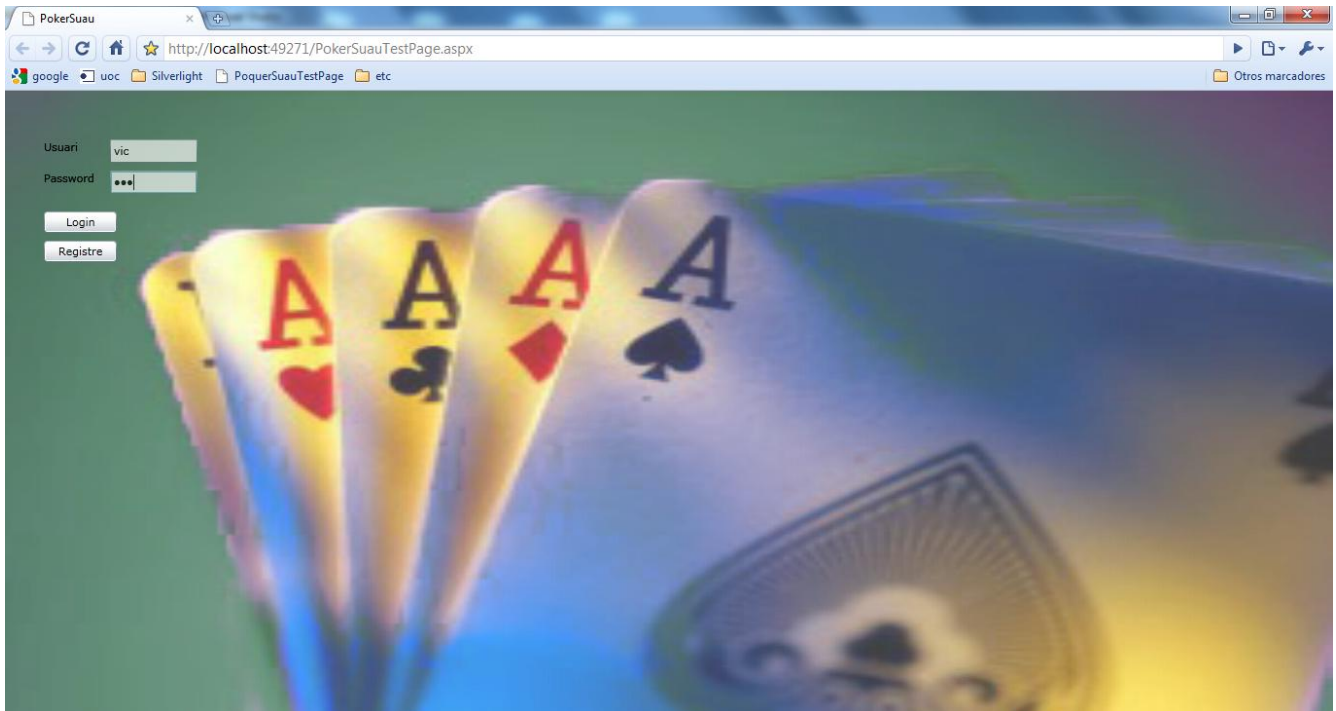
A continuació es mostra el diagrama de la base de dades definitiu:



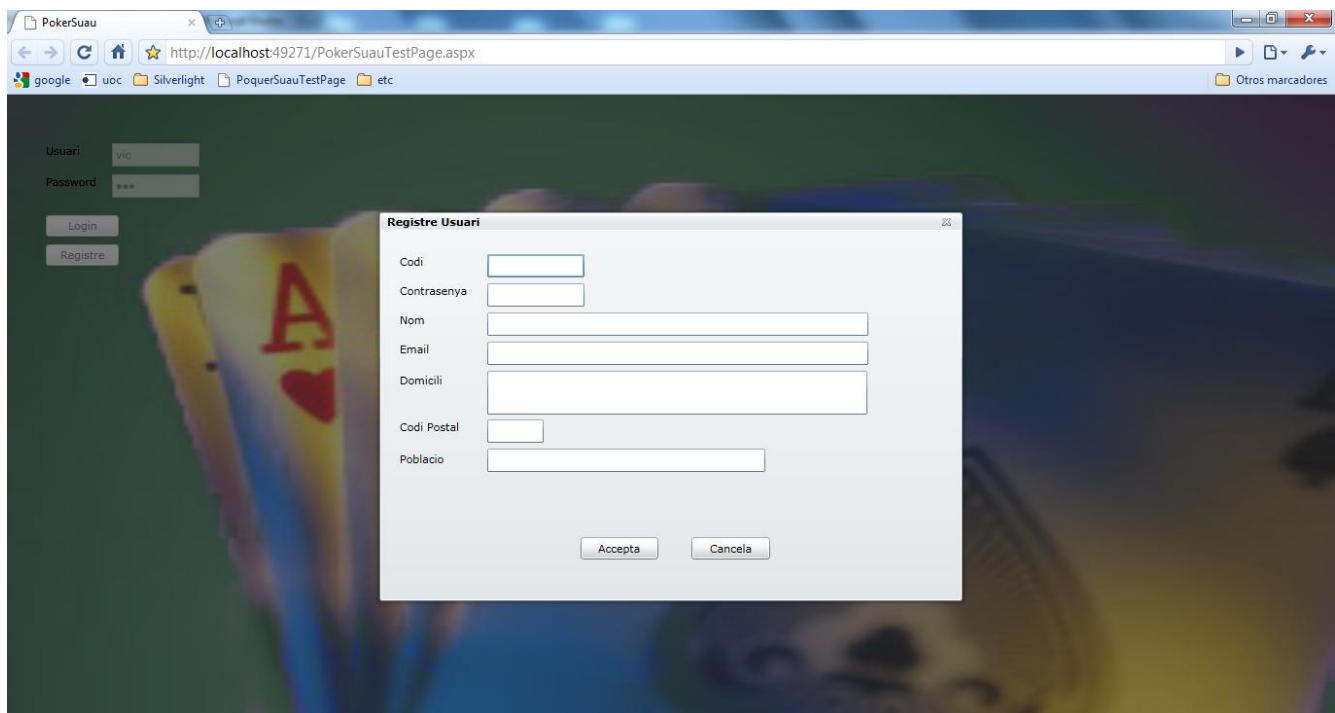
## 4.2. Interfície gràfica

A continuació s'exposen algunes imatges corresponents a diferents pantalles de la interfície gràfica.

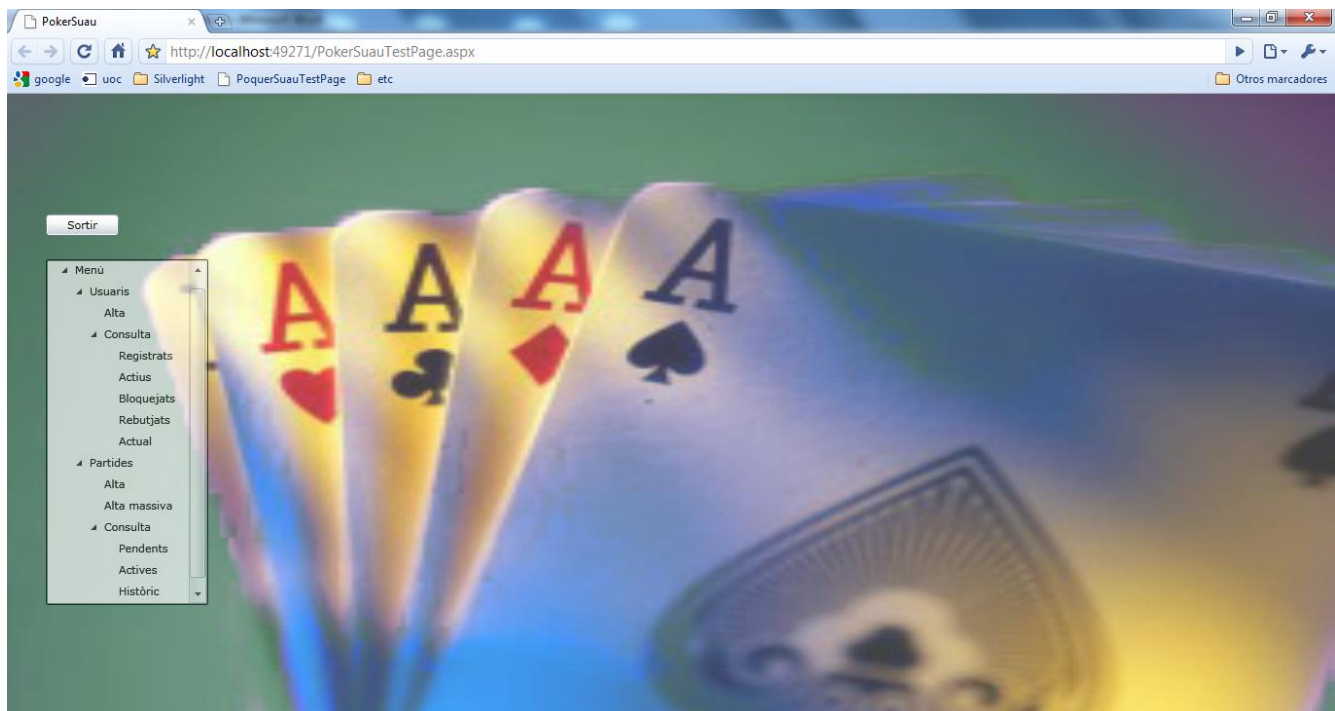
La pantalla principal, des de la qual s'ofereix a un usuari anònim la possibilitat de registrar-se, i el login a aquells usuaris que ja estan actius al sistema.



La figura següent mostra el formulari de registre pels usuaris anònims.



La figura següent representa el menú de l'administrador, un cop autenticat. Com es pot apreciar, totes les opcions de menú es disposen com a nodes en arbre.



Com opcions exclusives de l'administrador, existeix la següent pantalla que permet donar d'alta o modificar les dades d'un usuari.

**MantenimentUsuari**

Codi:

Contrasenya:

Nom:

Email:

Domicili:

Codi Postal:

Població:

Credit:

☐ Registrat
 ☐ Administrador

☒ Actiu
 ☐ Bloquejat
 ☐ Rebutjat

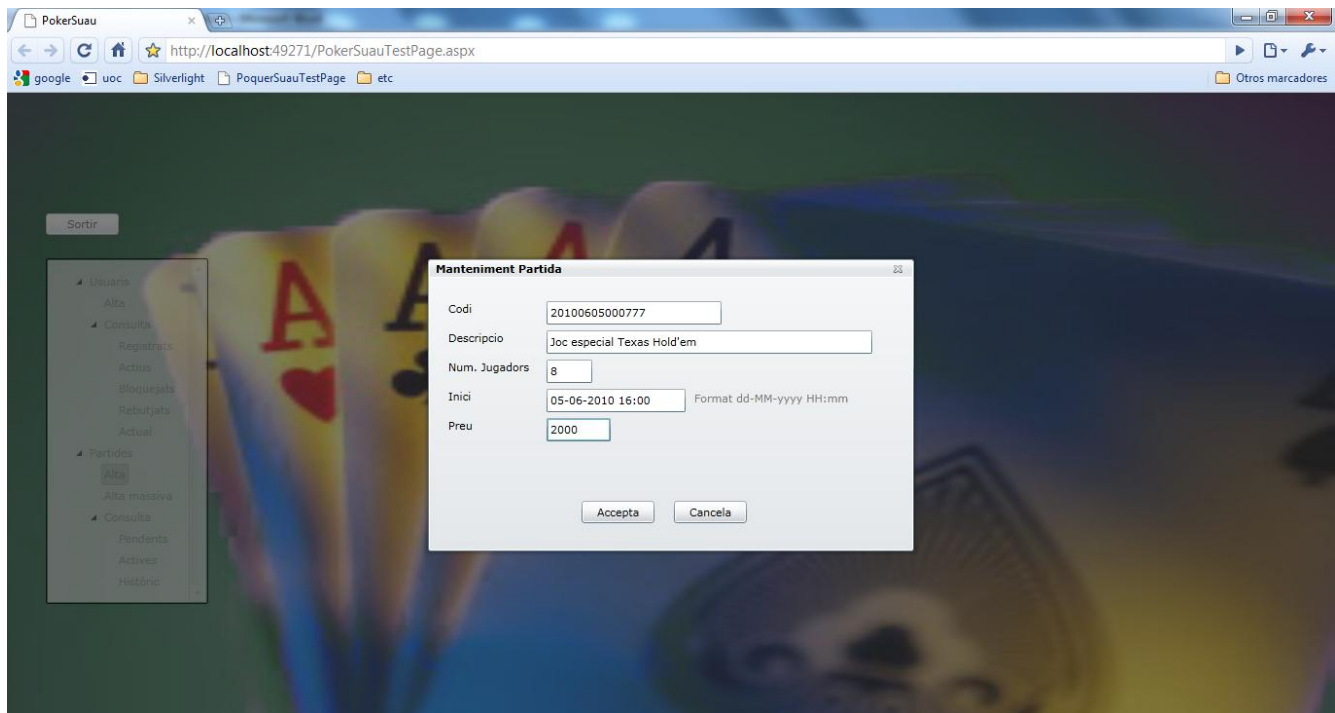
La següent figura mostra els usuaris actius.

**Actius**

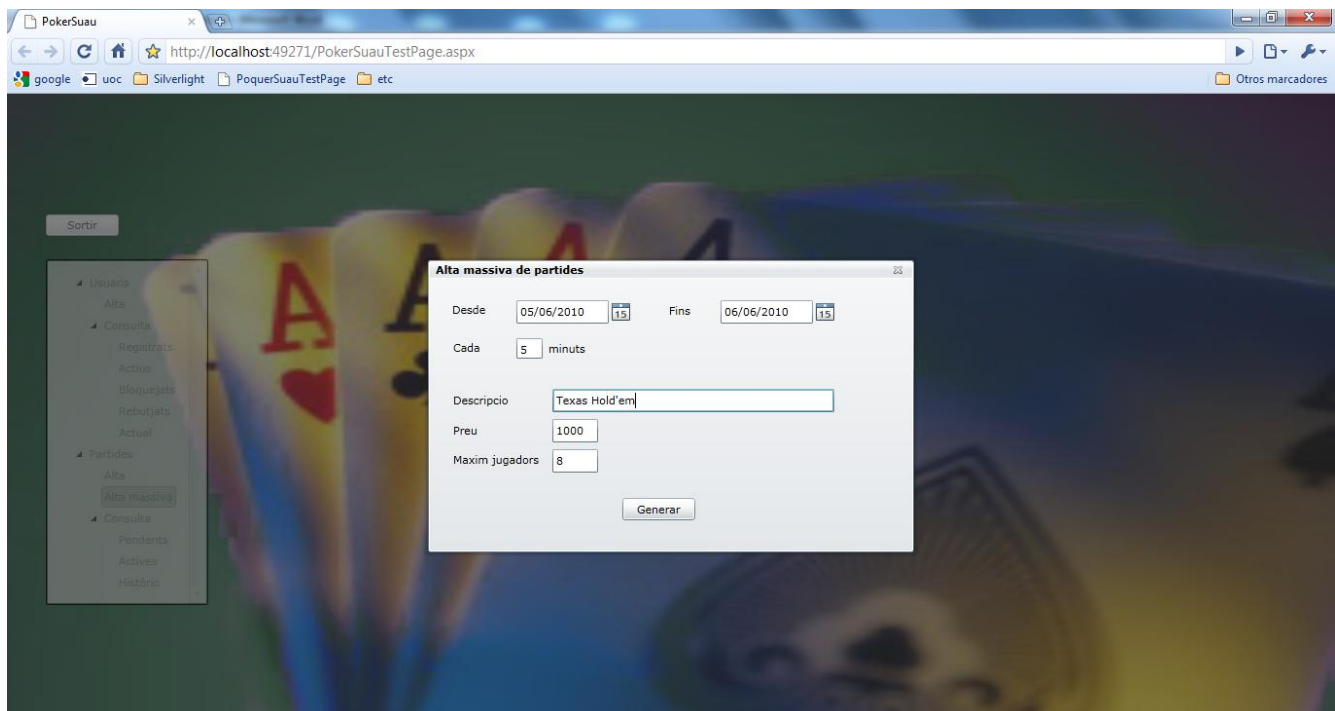
Usuari	Nom	Email	Població	Credit
admin	admin	admin@poquersuau.com	Sabadell	25400
anna	anna	anahzg@hotmail.com		15000
busquets	Busquets		Barberà del Vallès	10000
ibra	Zlatan Ibrahimovic	ibra@fcb.net	Barcelona	20000
iniesta	Andrés Iniesta	iniesta@live.com	Barcelona	20000
messi	messi	messi@fcb.com	barcelona	20000
pujol	Carles Pujol	pujol@fcb.cat	Barcelona	20000
sergi	Sergi Pérez	sergi@gmail.com	Sabadell	9000
tato	Tato Gutiérrez	tato@hotmail.com	sesrrovires	20000
toni	toni			16000
toure	toure yaya	toure@fcb.net	Barcelona	20000
vic	Victor Martín	vmartinper@uoc.edu	Castellarnau	12000
vm	victor	vmartinper@gmail.com		13000
xavi	Xavi Hernandez	xavi@fcb.com	Terrassa	20000



El manteniment de partides també és exclusiu de l'administrador.



Com a alternativa a l'alta de partides individual, l'administrador compta amb una utilitat per crear partides massivament.



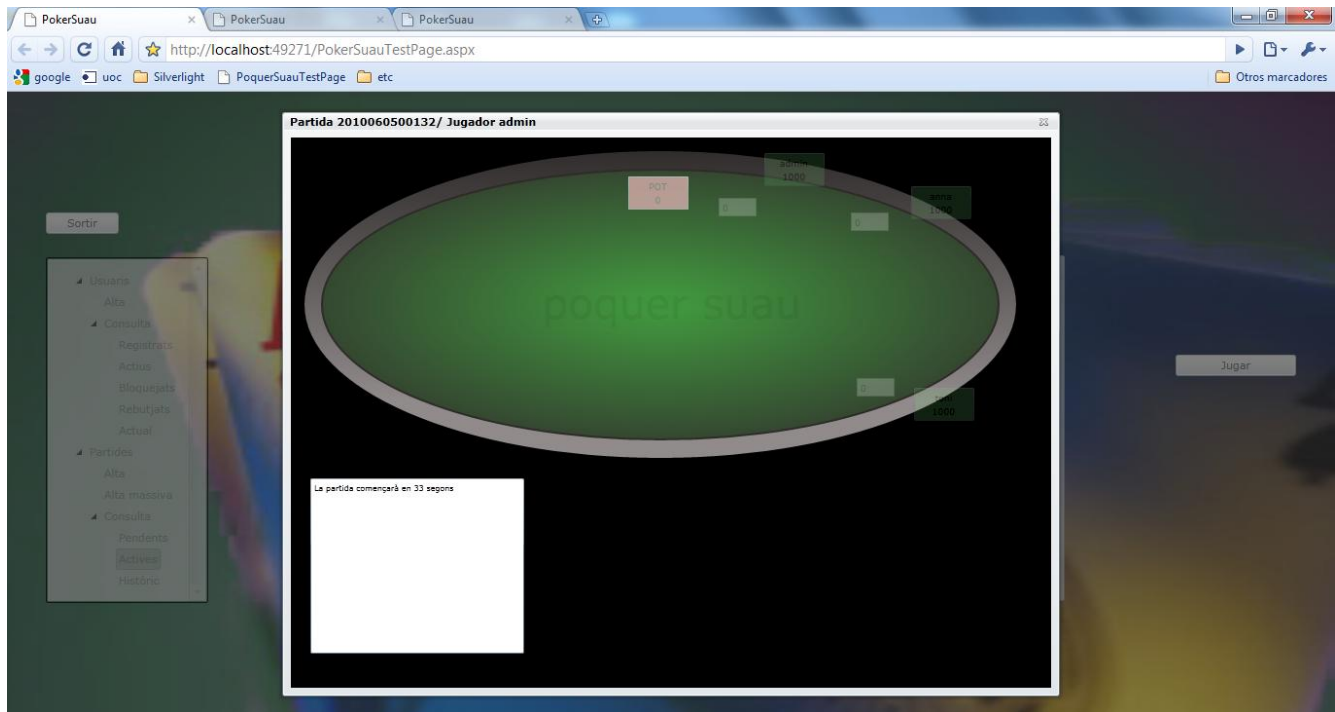
La imatge següent mostra les partides pendents. Aquesta consulta s'actualitza periòdicament tenint en compte l'hora actual del servidor.

Aquesta grid permet a l'usuari realitzar o cancel·lar la inscripció a una partida. També permet, només a l'administrador, accedir al manteniment de la partida.

The screenshot shows a web browser window with the URL `http://localhost:49271/PokerSuauTestPage.aspx`. The page displays a table of pending games. On the left, there is a navigation menu with options like 'Usuaris', 'Consulta', 'Partides', and 'Pendientes'. On the right, there are buttons for 'Edició', 'Inscripció', and 'Cancel·lar Inscripció'. The table lists 13 games, all of which are 'Texas Hold'em' with 8 players and a preu of 1000. The games are scheduled for June 5, 2010, at various times from 10:50 AM to 11:50 AM.

Partida	Descripció	Num.Jugadors	Inici	Preu	Inscripció
2010060500131	Texas Hold'em	8	6/5/2010 10:50:00 AM	1000	<input type="checkbox"/>
2010060500132	Texas Hold'em	8	6/5/2010 10:55:00 AM	1000	<input type="checkbox"/>
2010060500133	Texas Hold'em	8	6/5/2010 11:00:00 AM	1000	<input type="checkbox"/>
2010060500134	Texas Hold'em	8	6/5/2010 11:05:00 AM	1000	<input type="checkbox"/>
2010060500135	Texas Hold'em	8	6/5/2010 11:10:00 AM	1000	<input type="checkbox"/>
2010060500136	Texas Hold'em	8	6/5/2010 11:15:00 AM	1000	<input type="checkbox"/>
2010060500137	Texas Hold'em	8	6/5/2010 11:20:00 AM	1000	<input type="checkbox"/>
2010060500138	Texas Hold'em	8	6/5/2010 11:25:00 AM	1000	<input type="checkbox"/>
2010060500139	Texas Hold'em	8	6/5/2010 11:30:00 AM	1000	<input type="checkbox"/>
2010060500140	Texas Hold'em	8	6/5/2010 11:35:00 AM	1000	<input type="checkbox"/>
2010060500141	Texas Hold'em	8	6/5/2010 11:40:00 AM	1000	<input type="checkbox"/>
2010060500142	Texas Hold'em	8	6/5/2010 11:45:00 AM	1000	<input type="checkbox"/>
2010060500143	Texas Hold'em	8	6/5/2010 11:50:00 AM	1000	<input type="checkbox"/>

A continuació es mostren algunes imatges corresponents a l'àrea de joc. Com es pot apreciar, es tracta d'una finestra modal.

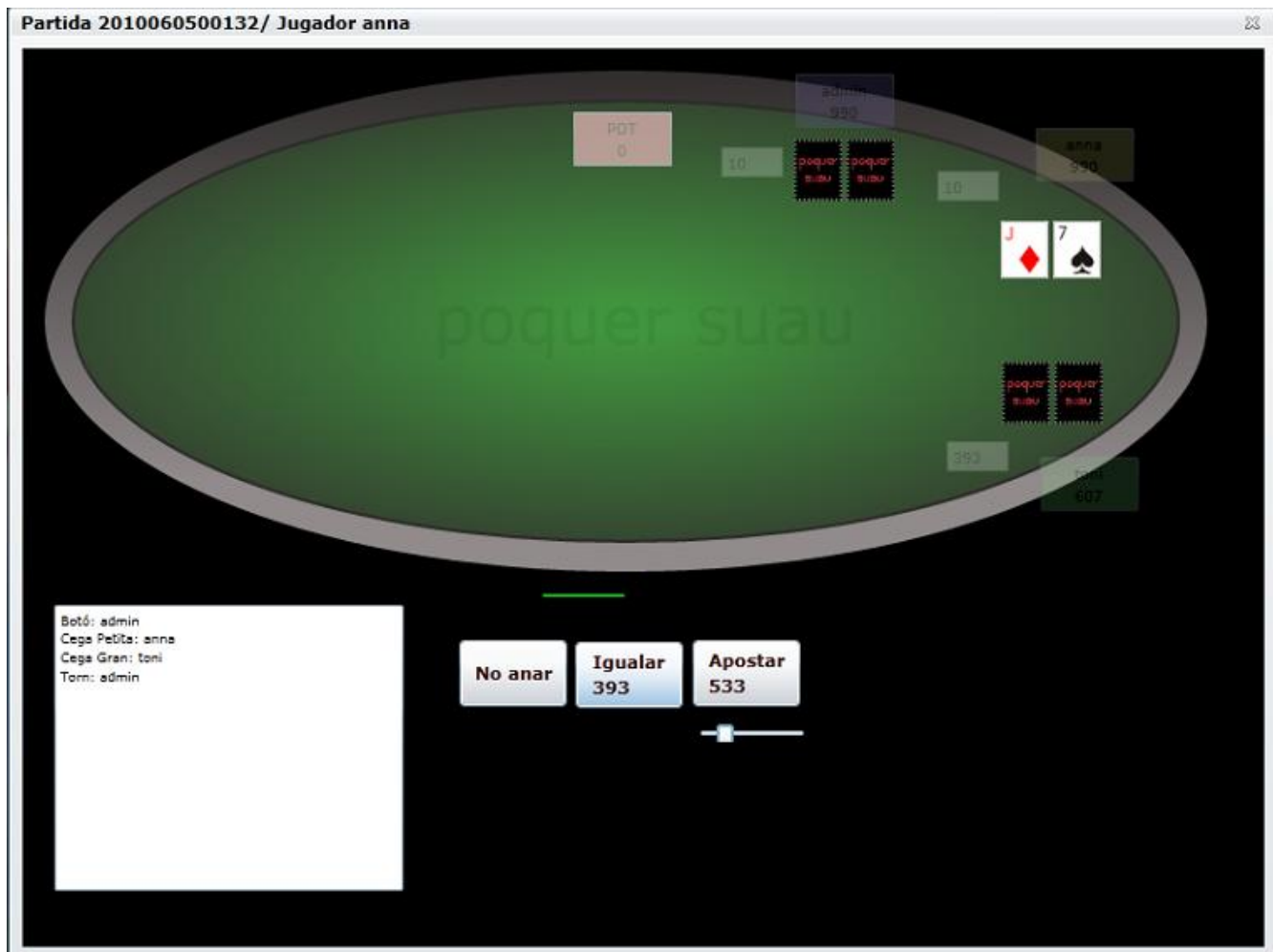




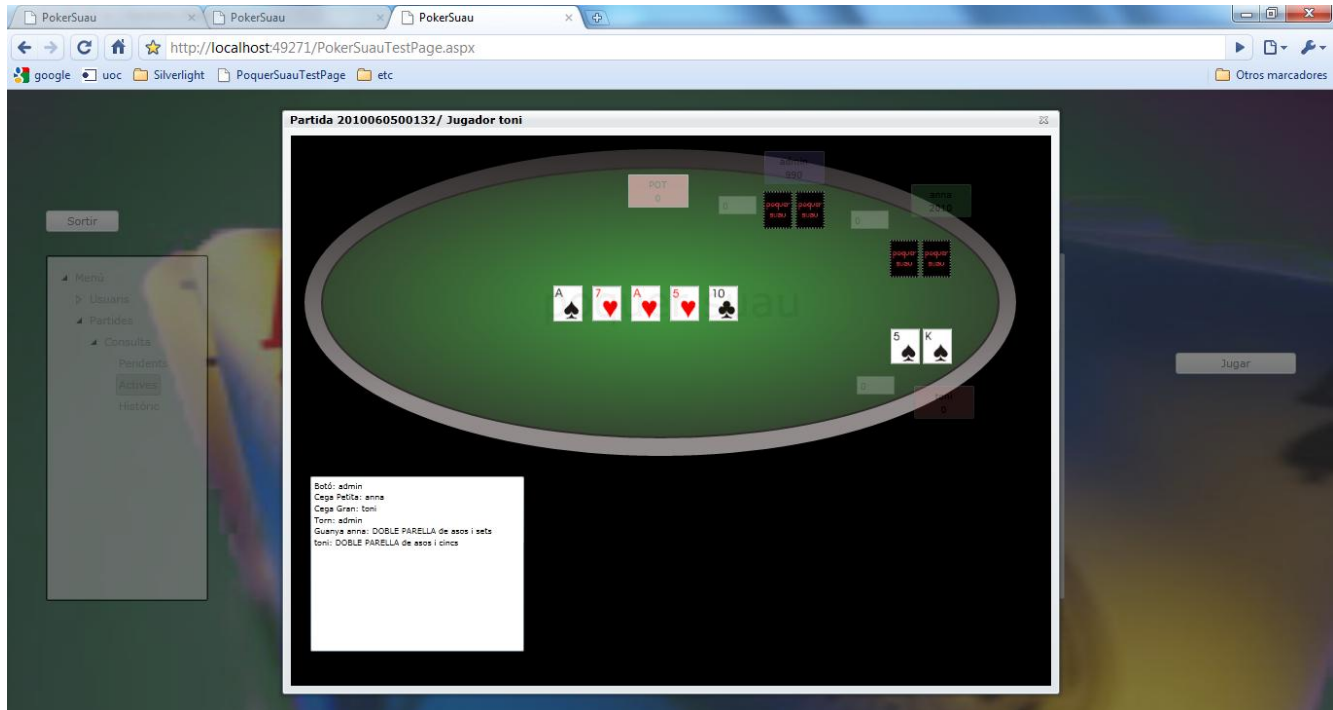
Com es pot observar, la imatge següent mostra a l'àrea de joc tres jugadors a una partida que està a punt de començar, i tots tres disposen inicialment del mateix crèdit.



La imatge següent il·lustra un altre instant de la partida, en que un dels jugadors gestiona el seu torn de joc.



La figura següent mostra el moment de resolució d'una mà, indicant les combinacions aconseguides per tots els jugadors que han cobert totes les apostes, arribant al final de l'esmentada mà, i assignant el pot al guanyador.



## 5. Línies de treball futur

Durant la implementació, vaig pensar que l'aplicació obtindria un important guany si l'àrea de joc estigués albergada en una finestra no modal. Després de visitar diversos fòrums, vaig veure que la plataforma Silverlight no proveeix aquesta característica, tot i que sembla que hi ha una important demanda i de ben segur tindria una bona acceptació entre els desenvolupadors. Per tant, en cas que en properes versions de Silverlight s'incorporés aquesta característica, una important línia de desenvolupament futur seria aquesta.

Un altre tret que faria més atractiva l'aplicació seria la incorporació d'un xat dins i fora de l'àrea de joc, de manera que els jugadors poguessin comunicar-se.

Respecte al manteniment de partides, seria interessant també que l'administrador pogués comptar amb una eina capaç d'esborrar les partides caducades on no s'ha inscrit cap jugador.

De cara a mantenir la base de dades i permetre que els accessos a la taula Partida siguin més ràpids, es pot afegir una eina que faci periòdicament un traspàs de partides antigues cap a una nova taula d'històric.

## 6. Conclusions

Aquest projecte ha estat un gran repte per mi. El caràcter asíncron de l'aplicació ha significat per mi moltes hores de dedicació, especialment pel que fa al desenvolupament del propi joc. A més, el fet de no haver dissenyat ni implementat mai una aplicació d'aquest tipus, sumat al meu desconeixement inicial de tecnologies com Silverlight, LinQ, serveis WCF, ... ha suposat per mi un esforç important.

No obstant sí he de destacar que m'ha estat molt còmode treballar amb les llibreries del Framework 3.5 de .NET, i també m'ha agradat molt el llenguatge utilitzat a la implementació, C#, llenguatge que també era desconegut per mi abans de començar el desenvolupament. També he descobert l'alt potencial que ofereix Silverlight per crear entorns d'usuari amb contingut multimèdia. Respecte LinQ, tot i ser una eina per estalviar temps d'implementació, m'ha semblat una mica massa tancat i en moltes ocasions m'he sentit més còmode gestionant molts dels processos del servidor mitjançant procediments emmagatzemats i desencadenadors des de l'SQL Server directament.

En resum, després de moltes hores de dedicació, puc dir que aquest projecte ha significat un valor afegit molt gran per mi i crec que he assolit els objectius fixats.

## 7. Glossari

### **AJAX (Asynchronous JavaScript And XML)**

És una tècnica de desenvolupament web per crear aplicacions interactives o RIA (Rich Internet Applications). Aquestes aplicacions s'executen al navegador mentre es manté la comunicació asíncrona amb el servidor en segon pla. D'aquesta manera és possible realitzar canvis sobre les pàgines sense necessitat de recarregar tota la pàgina des del servidor, cosa que fa augmentar la velocitat de les aplicacions.

AJAX és una tècnica vàlida per a múltiples plataformes i utilitzable en molts sistemes operatius i navegadors ja que està basat en estàndards oberts com JavaScript i DOM.

### **DOM (Document Object Model)**

El Model d'Objectes del Document és una interfície de programació d'aplicacions (API) per a documents HTML i XML. Defineix l'estructura lògica dels documents i la manera com s'accedeix i es manipula un document. A la interfície DOM, el document pot representar moltes classes diferents d'informació, i fa possible que els programadors puguin accedir a qualsevol element XML, modificar-lo, etc.

### **IIS (Internet Information Server)**

És un servidor que ofereix un conjunt de serveis web dins de la plataforma Windows.

### **Microsoft Silverlight**

És un subconjunt de WPF i està orientat a aplicacions web dins de la plataforma .NET. És un complement per a navegadors d'Internet basat en la plataforma Windows que afegeix noves funcions multimèdia. La base de la seva programació és XAML i accedeix als objectes per mitjà de C #, Visual Basic i JavaScript.

Microsoft Expression Blend i Microsoft Expression Designer són eines de Microsoft que s'utilitzen per crear les animacions en Silverlight.

Silverlight permet carregar dinàmicament un contingut XML que pot ser manipulat a través d'una interfície DOM, una tècnica que és compatible amb les convencionals del llenguatge AJAX.

### **WCF (Windows Communication)**

És un model de programació unificat de Microsoft per generar aplicacions orientades a serveis. Permet que les aplicacions client-servidor puguin realitzar transaccions segures entre diferents plataformes.

## **WPF (Windows Presentation Foundation)**

És un subsistema per la creació d'aplicacions basades en Windows. Proporciona una separació entre la interfície d'usuari i la lògica de negoci.

WPF també proveeix un llenguatge de marcat, XAML, que serveix com a eina alternativa per a la definició d'elements de la interfície i les relacions amb altres elements de la interfície d'usuari.

## **XAML (eXtensible Application Markup Language)**

És el llenguatge de format per a les interfícies gràfiques d'usuari WPF i Silverlight.

Es tracta d'un llenguatge declaratiu basat en XML, optimitzat per a definir interfícies gràfiques d'usuari de gran atractiu.

## 8. Bibliografia

### Llibres

Silverlight 2.0. Laurence Moroney. Anaya Multimedia.

### Referències d'Internet

MSDN Library

<http://msdn.microsoft.com>

ASP.NET

<http://thinkingindotnet.wordpress.com>

Silverlight

<http://www.silverlight.net>

C#

<http://www.koders.com/zeitgeist/csharp/>

Cercador Google

[www.google.com](http://www.google.com)

Wikipedia

<http://www.wikipedia.org>