

TFC .NET - Hotel Spa

Memòria

Hotel SPA. Aplicació desenvolupada amb tecnologia .NET per a un hotel amb diferents instal·lacions d'esbarjo i relaxació.

Alumne: Manel Bañó Richart

Consultor: Jordi Sánchez Cano

Estudis d'Enginyeria Tècnica d'Informàtica de Sistemes

Any acadèmic 2009/2010, 2on semestre

Data: 11 de juny de 2010

“A la meva parella, Ana, que sense ella no hagués estat possible. Gràcies a ella vaig començar la carrera que ara finalitzo.”

Índex

Índex	3
Índex Figures	5
Resum	6
1. Introducció.....	7
1.1. Definició del projecte: l'escenari de partida.....	7
1.2. Justificació del projecte.....	8
1.3. Descripció del projecte.....	8
1.4. Objectius	9
1.5. Planificació del projecte.....	10
1.5.1. Temporització del projecte.....	11
1.5.2. Cicle de vida.....	13
1.5.3. Tasques realitzades.....	14
1.6. Tecnologies utilitzades.....	15
1.6.1. Programari utilitzat.....	17
1.7. Productes obtinguts	17
2. Requeriments inicials	19
2.1. Requeriments funcionals.....	19
2.2. Requeriments no funcionals.....	20
3. Anàlisi del sistema.....	21
3.1. Diagrama de casos d'ús.....	21
3.1.1. Diagrama de casos d'ús Accedir al Sistema.....	21
3.1.2. Diagrama de casos d'ús General	22
3.1.3. Diagrama de casos d'ús Gestió Usuaris	22
3.1.4. Diagrama de casos d'ús Gestió Serveis.....	24
3.1.5. Diagrama de casos d'ús Gestió Recursos.....	25
3.1.6. Diagrama de casos d'ús Gestió Activitats	26
3.2. Especificació textual dels casos d'ús anteriors.....	27
3.2.1. Casos d'ús Accedir Sistema	27
3.2.2. Casos d'ús Gestió Usuaris.....	27
3.2.3. Casos d'ús Gestió Serveis	29
3.2.4. Casos d'ús Gestió Recursos	32

3.2.5. Casos d'ús Gestió Activitats	35
3.3. Model Conceptual	39
3.4. Diagrama de classes d'entitat	39
3.4.1. Reificació de les classes associatives binàries i de les associacions n-àries	39
3.4.2. Altres transformacions	40
3.4.3. Diagrama de Classes	40
3.5. Altres diagrames.....	41
3.5.1. Diagrama de seqüència Login	41
3.5.2. Diagrama de seqüència d'una Inscripció a una Activitat	42
3.5.3. Diagrama de seqüència Alta Servei.....	42
4. Disseny	43
4.1. Arquitectura de l'aplicació	43
4.1.1. Arquitectura física.....	43
4.1.2. Arquitectura lògica.....	44
4.2. Diagrama de desplegament	44
4.3. Disseny Base de Dades	45
4.3.1. Obtenció del disseny de les taules.....	45
4.3.2. Diagrama de la base de dades.....	47
4.4. Classes de LINQ-To-SQL.....	48
5. Implementació	49
5.1 Etapes	49
6. Treball futur	50
7. Conclusions	51
Bibliografia.....	53

Índex Figures

Figura 1. Temporització del projecte.....	11
Figura 2. Diagrama de Gantt.	12
Figura 3. Cicle de vida en cascada.	13
Figura 4. Arquitectura d'aplicacions .NET.	16
Figura 5. Cas d'ús Accedir Sistema	21
Figura 6. Cas d'ús general del sistema.	22
Figura 7. Cas d'ús Gestió Usuaris.	23
Figura 8. Cas d'ús Gestió Serveis.....	24
Figura 9. Cas d'ús Gestió Recursos.....	25
Figura 10. Cas d'ús Gestió Activitats.....	26
Figura 11. Classes d'entitat Model Conceptual.	39
Figura 12. Reificació de classe associativa.	39
Figura 13. Diagrama de classes.	40
Figura 14. Diagrama de seqüència Login.....	41
Figura 15. Diagrama de seqüència Inscripció Activitat.	42
Figura 16. Diagrama de seqüència Alta Servei.	42
Figura 17. Arquitectura física.....	43
Figura 18. Diagrama de desplegament.	45
Figura 19. Diagrama base de dades.....	47
Figura 20. Diagrama de classe Linq To Sql.	48

Resum

El present Treball de Fi de Carrera ha consistit en realitzar una aplicació per un hotel amb diferents instal·lacions d'esbarjo i relaxació utilitzant la tecnologia .NET de Microsoft.

L'aplicació resultant permet als clients de l'hotel poder informar-se sobre els serveis, activitats i recursos que estan disponibles i poder fer les peticions, inscripcions o reserves que els interessin. Per altra banda, els administradors, a part de poder gestionar tota aquesta informació, poden gestionar els usuaris.

La informació està emmagatzemada en un servidor de bases de dades Microsoft SQL Server 2008. L'aplicació d'escriptori, realitzada amb WPF (Windows Presentation Foundation), es comunicarà amb un servei WCF (Windows Communications Foundation), que serà l'encarregat de treballar amb les dades de la base de dades mitjançant una biblioteca de classes creada amb LINQ to SQL.

L'entorn de treball utilitzat ha estat el Microsoft Visual Studio 2008 SP1 juntament amb el Framework 3.5 de .NET.

Aquest treball ha estat realitzat en 4 fases:

- ✓ Pla de Treball.
- ✓ Anàlisi i Disseny.
- ✓ Implementació.
- ✓ Memòria i presentació virtual.

L'arquitectura del sistema està dividida en 3 capes anomenades 'Capa de presentació', 'Capa de lògica de negoci' i 'Capa d'accés a dades'. El desenvolupament en capes ens permet separar l'accés a les dades de la lògica de negoci o de la presentació i, d'aquesta forma, la modificació d'alguna d'aquestes capes no interfereix en el codi de la resta de capes.

Aquest Treball de Fi de Carrera de desenvolupament d'una aplicació amb Microsoft .NET ha sigut la meva primera experiència amb aquesta tecnologia i m'ha permès introduir-me al llenguatge dissenyat expressament per utilitzar la tecnologia .NET, el C#, descobrir WPF i XAML per dissenyar interfícies d'usuari intuïtives i potents, conèixer els serveis WCF, veure la facilitat que ens dona LINQ To SQL per crear les classes d'entitat de les taules de la base de dades o utilitzar per primera vegada Microsoft SQL Server 2008.

1. Introducció

1.1. Definició del projecte: l'escenari de partida

L'enunciat de partida escollit entre els diferents que se'ns presentaven ha estat el següent:

Hotel SPA

Es tracta de desenvolupar una aplicació per a un hotel amb diferents instal·lacions d'esbarjo i relaxació.

Cada habitació de l'hotel tindrà un terminal que permetrà als clients accedir a les diferents opcions del sistema. Aquests terminals s'ubicaran també en altres ubicacions de l'hotel per fer més fàcil el seu accés en qualsevol moment.

A través dels terminals, els clients tindran disponibles diverses funcionalitats, com per exemple (però no limitat a):

- ✓ *peticions al servei d'habitacions (incidències o peticions de material, servei de despertador, servei de neteja, etc),*
- ✓ *reservar recursos (com ara sales de reunions, pistes d'esport, etc.),*
- ✓ *planificar activitats realitzades al recinte de l'hotel (gimnàs, massatgista, conferències, audicions musicals o de vídeo, etc)*

Es convenient comentar que les activitats i els recursos poden tenir places limitades i per tant cal gestionar els seus respectius calendaris d'ocupació.

Per altra banda, els encarregats de l'hotel disposaran d'una aplicació per controlar i gestionar totes les peticions dels clients, per tal de donar les ordres adients per a satisfer les seves demandes.

Les aplicacions client estaran desenvolupades en WPF (Windows Presentation Foundation). La lògica de negoci del sistema estarà implementada en un servidor dedicat en el que també s'allotjarà la base de dades. La comunicació entre les interfícies i la lògica de negoci es farà mitjançant WCF (Windows Communications Foundation).

1.2. Justificació del projecte

L'hotel SPA vol tenir un software que permeti als seus clients gaudir, de forma interactiva, de les seves instal·lacions i serveis. L'usuari podrà utilitzar les instal·lacions de l'hotel i reservar plaça sense tenir que anar a la recepció per informar-se, ja que l'aplicació li donarà tota la informació necessària.

Al mercat existeixen diferents aplicacions per a la gestió d'hotels però no hem trobat cap amb la que es permeti al propi client accedir a l'aplicació i poder actuar directament sobre ella. Aquest és un fet diferencial que podrem donar a l'hotel i suposarà un valor afegit al client.

Per tant, permetre que el client sigui qui consulti directament als terminals la informació de la que estigui interessat des de la seva pròpia habitació o des de altres terminals instal·lats a l'hotel, realitzar peticions, inscripcions o reserves o anul·lar-les, dona una immediatesa que serà ben rebuda pels usuaris ja que no tindran que anar a la recepció de l'hotel i esperar que el recepcionista els atengui.

1.3. Descripció del projecte

Aquest treball té per finalitat l'anàlisi, disseny i implementació d'una aplicació utilitzant les eines que ens ofereix la tecnologia .NET.

Per una banda tenim una base de dades, *HSPA.mdf*, realitzada amb Microsoft SQL Server 2008 amb totes les dades necessàries pel correcte funcionament de l'aplicació. Aquesta base de dades es trobarà instal·lada a un servidor on es connectarà l'aplicació WCF.

HotelSpaWCF és el servei que serà utilitzat per l'aplicació client, desenvolupada amb WPF, per treballar amb les dades. Aquest servei WCF fa ús de la llibreria de classes *LinqHotelSpa.dll* que és l'encarregada de convertir les taules de la base de dades en classes de C#. El servei ofereix uns mètodes a l'aplicació WPF que li permeten crear, consultar, modificar o eliminar dades de la base de dades. Aquest servei també es trobarà instal·lat al servidor i s'utilitzarà una petita aplicació de consola, *Servidor.exe*, per hostatjar-lo.

WpfHotelSpa és l'aplicació client que permetrà als usuaris i administradors, treballar sobre la informació de la base de dades.

L'aplicació donarà diferents tipus de serveis als clients de l'hotel, com per exemple:

- ✓ Obtenir informació sobre els serveis oferts, les activitats o els recursos disponibles.
- ✓ Realitzar peticions dels serveis (neteja, despertador, bar, etc.), reservar recursos com sales de reunions, pistes d'esport, etc., o inscriure's a les activitats que es realitzen al recinte.
- ✓ Els encarregats (administradors) gestionaran les peticions, inscripcions i reserves dels clients. També gestionaran els serveis, activitats i recursos així com els usuaris.

El client podrà fer les seves peticions mitjançant terminals instal·lats a les habitacions i, a altres punts de les instal·lacions que faran més fàcil el seu accés en qualsevol moment.

Per accedir a l'aplicació serà necessari introduir un nom d'usuari i una contrasenya que, un cop validats, donaran accés a la part de l'aplicació corresponent als usuaris (és l'aplicació que utilitzaran els clients de l'hotel) o, si l'usuari introduït és un administrador, a la part de l'aplicació corresponent als administradors (és l'aplicació de gestió que utilitzaran els responsables de l'hotel).

1.4. Objectius

Com s'ha comentat abans al resum, els objectius del programari són dotar aquest hotel dels avantatges que ofereix aquesta aplicació i donar als seus clients el millor servei possible per aprofitar les seves instal·lacions.

D'altra banda, existeixen uns altres objectius, els acadèmics. Aquests objectius són els següents:

- ✓ Consolidar els coneixements adquirits al llarg dels estudis d'Enginyeria Tècnica d'Informàtica de Sistemes.
- ✓ Conèixer i aprendre a utilitzar la plataforma de desenvolupament .NET de Microsoft.
- ✓ Utilitzar la tecnologia WPF(*Windows Presentation Foundation*), WCF(*Windows Communications Foundation*) i LINQ que ens ofereix el Framework 3.5 de .NET.

- ✓ Utilitzar el sistema gestor de bases de dades Microsoft SQL Server 2008.
- ✓ Crear l'aplicació utilitzant el *desenvolupament en tres capes*: capa de presentació, capa de negoci i capa de dades.

1.5. Planificació del projecte

Al començament del semestre es va realitzar una planificació del projecte, dins del document “Pla de Treball”, on es realitzava una planificació temporal de les tasques a realitzar així com de les tecnologies a utilitzar. Aquesta planificació temporal especificava les fites que volíem aconseguir, així com la duració i la data d'entrega de cada PAC.

1.5.1. Temporització del projecte

El calendari següent va ser el següent:

Nombre de tarea	Duración	Comienzo	Fin	Predecesoras
INICI DEL PROYECTE	0 días	jue 25/02/10	jue 25/02/10	
☐ PAC1 - Pla de Treball	12 días	jue 25/02/10	mié 10/03/10	
Descripció del treball	2 días	jue 25/02/10	vie 26/02/10	
Objectius	1 día	sáb 27/02/10	sáb 27/02/10	3
Estudi de la idoneïtat del projecte	2 días	dom 28/02/10	lun 01/03/10	4
Anàlisi de requeriments informal	2 días	mar 02/03/10	mié 03/03/10	5
Tecnologies a utilitzar	1 día	jue 04/03/10	jue 04/03/10	6
Tasques i recursos necessaris	2 días	vie 05/03/10	lun 08/03/10	7
Localització del software necessari	1 día	mar 09/03/10	mar 09/03/10	8
Planificació (Gantt)	3 días	lun 08/03/10	mié 10/03/10	
☐ PAC2 - Anàlisi i disseny	22 días	jue 11/03/10	mié 07/04/10	2
Instal·lació software d'anàlisi, dis	1 día	jue 11/03/10	jue 11/03/10	
Anàlisi prèvia i de requisits	4 días	vie 12/03/10	mié 17/03/10	13
Model dinàmic: Diagrama de caso	3 días	jue 18/03/10	sáb 20/03/10	14
Model estàtic: Diagrama de class	5 días	dom 21/03/10	jue 25/03/10	15
Altres diagrames	3 días	vie 26/03/10	mar 30/03/10	16
Disseny Base de Dades	2 días	mié 31/03/10	jue 01/04/10	17
Preparació i realització entrega	2 días	vie 02/04/10	lun 05/04/10	18
Estudi de la tecnologia de desenvol	22 días	jue 11/03/10	mié 07/04/10	
☐ PAC3 - Implementació	34 días	jue 08/04/10	mar 25/05/10	12
Prototips d'interfícies (WPF)	3 días	jue 08/04/10	lun 12/04/10	
Base de Dades (SQL Server 200	4 días	mar 13/04/10	vie 16/04/10	23
Connexió amb la BBDD (WCF)	2 días	lun 19/04/10	mar 20/04/10	24
Implementació (C#)	18 días	mié 21/04/10	vie 14/05/10	25
Proves	5 días	lun 17/05/10	vie 21/05/10	26
Preparació i realització entrega	2 días	lun 24/05/10	mar 25/05/10	27
☐ Final - Memòria i presentació vir	13 días	mié 26/05/10	vie 11/06/10	22
Memòria	5 días	mié 26/05/10	mar 01/06/10	
Presentació (Power Point)	6 días	mié 02/06/10	mié 09/06/10	31
Implementació definitiva	2 días	jue 10/06/10	vie 11/06/10	32
Debat Virtual	4 días	mar 22/06/10	vie 25/06/10	
FIDEL PROYECTE	0 días	vie 25/06/10	vie 25/06/10	

Figura 1. Temporització del projecte.

I el corresponent diagrama de Gantt:

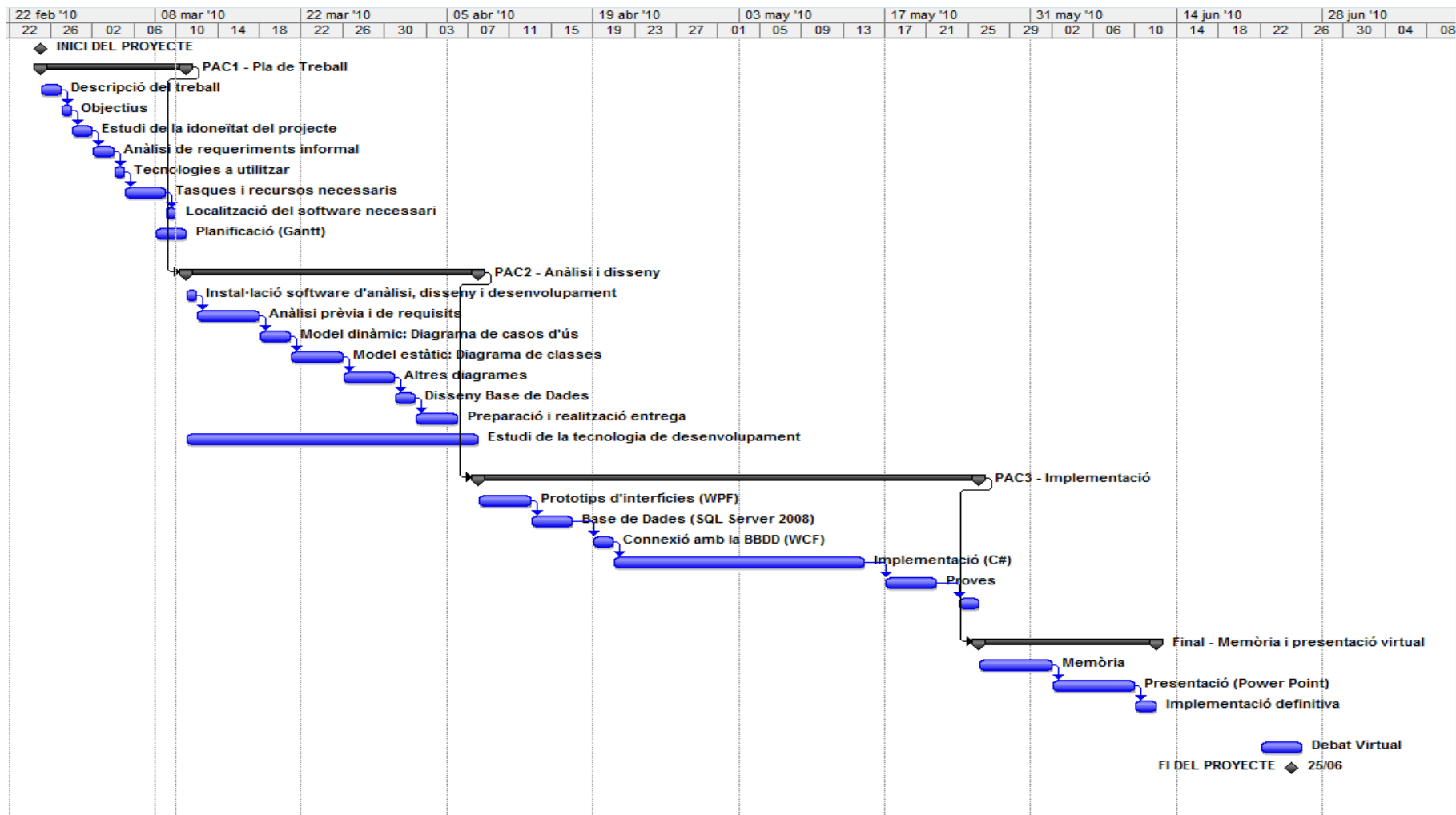


Figura 2. Diagrama de Gantt.

Encara que les fites realitzades i la documentació generada en cada fase han sigut lliurades en els terminis establerts al pla de treball, la duració de cada tasca individual no ha estat del tot exacte ja que, sobretot a la part d'implementació, he tingut que cercar molta informació, especialment sobre C#, WCF i WPF, per poder mostrar la informació de la forma que tenia pensada en un principi. Per tant, la tasca "Estudi de la tecnologia de desenvolupament" va començar el dia establert però ha durat pràcticament fins a l'entrega de la PAC3 – Implementació.

En aquesta memòria es resumeixen els continguts de les diferents fases.

1.5.2. Cicle de vida

El projecte s'ha desenvolupat seguint el cicle de vida clàssic de desenvolupament de programari, lleugerament adaptat als requeriments del nostre projecte:

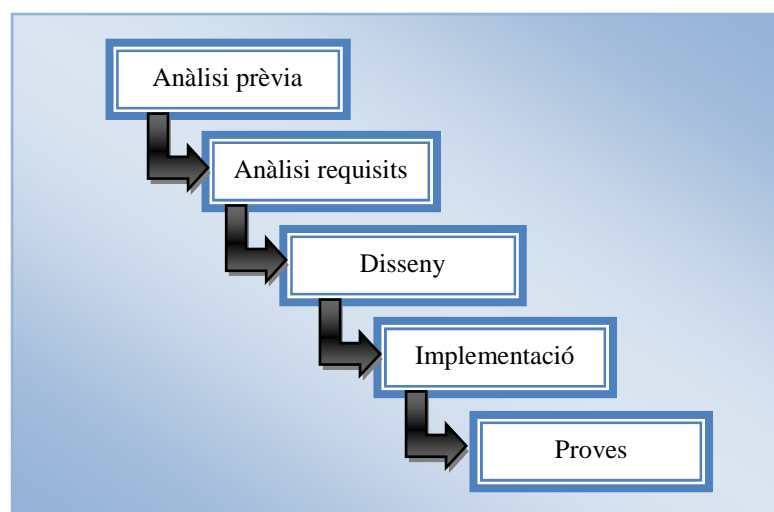


Figura 3. Cicle de vida en cascada.

1.5.3. Tasques realitzades

En aquest projecte s'han identificat clarament 5 tasques. Aquestes coincideixen exactament amb les 3 Pac's, l'entrega de la Memòria + presentació virtual del projecte i el debat virtual:

- ✓ **PAC1: Pla de treball.** Període del 25 de Febrer al 10 de Març de 2010.
 - ✓ En aquesta tasca hem fet un estudi inicial sobre el projecte a desenvolupar així com una planificació per a dur a terme totes les tasques necessàries per finalitzar el projecte a la data proposada.

- ✓ **PAC2: Anàlisi i disseny.** Període del 11 de Març al 7 d'Abril de 2010.
 - ✓ Instal·lació del software necessari pel desenvolupament.
 - ✓ Aprendre a utilitzar la tecnologia .NET amb WPF,WCF,Linq-to-SQL i SQL Server.
 - ✓ Realitzar els diagrames dinàmics i estàtics de l'aplicació.
 - ✓ Disseny de la base de dades.

- ✓ **PAC3: Implementació.** Període del 8 d'Abril al 25 de Maig de 2010.
 - ✓ Realització dels prototips de l'aplicació.
 - ✓ Implementació de la base de dades i de l'aplicació.
 - ✓ Procés de proves.

- ✓ **Memòria i presentació virtual.** Període del 26 de Maig al 11 de Juny de 2010.
 - ✓ Preparació de la Memòria i la documentació del projecte.
 - ✓ Creació d'una presentació virtual de vídeo amb Power Point.
 - ✓ Implementació definitiva.

- ✓ **Debat virtual.** Període del 22 de Juny al 25 de Juny de 2010.
 - ✓ Debat virtual a l'aula amb els companys i el consultor del TFC .NET.

1.6. Tecnologies utilitzades

L'aplicació ha estat desenvolupada amb la tecnologia **.NET** de Microsoft. A partir de la versió 3.0 del Framework **.NET** s'integren dos components fonamentals per a la creació d'aplicacions amb aquesta tecnologia:

- ✓ **WPF**: *Windows Presentation Foundation*, inicialment anomenat "Avalon", separa amb el llenguatge declaratiu XALM, *eXtensible Application Markup Language*, i els llenguatges de programació de **.NET**, la interfície d'interacció de la lògica de negoci.

Amb XALM podem definir els elements que compondran la interfície d'usuari que podran ser connectats amb la lògica de negoci mitjançant el motor de presentació WPF. La lògica de l'aplicació pot ser programada amb qualsevol llenguatge **.NET**. En el nostre cas utilitzarem el llenguatge **C#** que va ser desenvolupat especialment per aquesta tecnologia.

- ✓ **WCF**: *Windows Communications Foundation*, conegut amb el nom intern "Indigo", és un sistema de comunicació basat en missatges sobre múltiples transports i a través de sistemes heterogenis.

Per l'aplicació client hem utilitzat WPF (*Windows Presentation Foundation*) que ens ha permès, per una banda, treballar amb XALM per al disseny de la interfície d'usuari i, per l'altre, programar la lògica de l'aplicació en **C#**.

Les comunicacions entre la interfície i la base de dades han estat realitzades amb WCF (*Windows Communications Foundation*). Aquest sistema és l'encarregat de dur a terme les comunicacions necessàries entre l'aplicació i la base de dades fent ús de la llibreria de classes LINQ To SQL creada a tal efecte.

El sistema gestor de bases de dades utilitzat per la nostra aplicació ha estat **Microsoft SQL Server 2008 Developer** i l'aplicació accedirà a la base de dades mitjançant **LINQ** (*Language Integrated Query*). Aquesta nova tecnologia desenvolupada per Microsoft a partir del Framework 3.5 de **.NET** ens proporciona una infraestructura en temps d'execució per administrar les dades relacionals com objectes. LINQ defineix operadors de consulta estàndard que permeten a llenguatges habilitats amb LINQ filtrar, enumerar i crear projeccions de diversos tipus de col·leccions utilitzant la mateixa sintaxis.

El framework LINQ inclou una eina anomenada *SQLMetal* que permet la generació automàtica de classes directament d'una base de dades MS-SQL. D'aquesta forma permet la integració fàcilment del codi de l'aplicació amb la base de dades de forma fàcil i ràpida. Concretament utilitzarem *LINQ-To-SQL*.

La tecnologia .NET ens permet treballar amb una arquitectura de desenvolupament de tres capes. Això significa que podem separar la lògica de negoci de l'accés a dades o de la presentació:

- ✓ **Capa de presentació:** Permet l'interacció usuari-sistema, que pot ser amb formularis Windows o Web.
- ✓ **Capa de lògica de negoci:** Permet la validació de dades i conté, al detall, el funcionament de l'aplicació.
- ✓ **Capa d'accés a dades:** S'encarrega d'interactuar amb la base de dades del sistema desenvolupat.

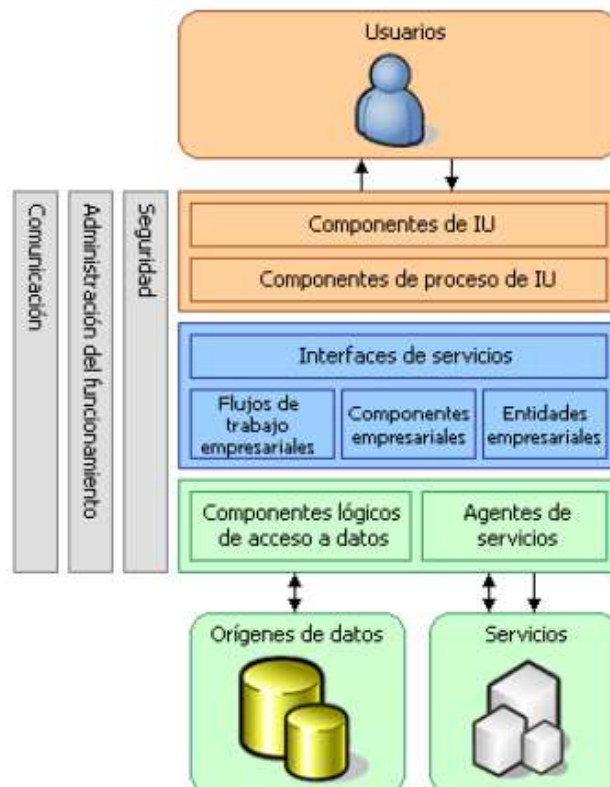


Figura 4. Arquitectura d'aplicacions .NET.

1.6.1. Programari utilitzat

Una de les tasques ha estat obtenir el software necessari per poder realitzar el projecte. Un cop cercat, instal·lat i provat, els programes que s'han utilitzat han sigut:

- ✓ Microsoft Project 2007, per planificar el projecte i crear el diagrama de Gantt.
- ✓ Magic Draw UML 16.0 SP1, per a la realització dels diagrames UML.
- ✓ Visual Studio 2008 SP1, pel desenvolupament de l'aplicació utilitzant el llenguatge C# i les tecnologies WPF, WCF i LINQ.
- ✓ Microsoft SQL Server 2008, com a sistema gestor de bases de dades.
- ✓ Microsoft SQL Server Management Studio, per dissenyar la base de dades.
- ✓ Microsoft Word i PDFCreator, el primer per elaborar la documentació i el segon per poder entregar-la en fitxers pdf.
- ✓ Camtasia Studio i Microsoft Power Point 2007, per a la realització de la presentació virtual.

Tot el software utilitzat de Microsoft s'ha obtingut de [MSDN Academic Alliance](#), web de Microsoft que col·labora amb diferents Universitats per a que el seus estudiants puguin obtenir el seu software de desenvolupament gratuïtament per treballar amb la tecnologia .NET.

1.7. Productes obtinguts

Els productes obtinguts en les diferents etapes han estat els següents:

- ✓ **Document del Pla de Treball, mbano_PlaTreball.pdf:**
Aquest document inclou la descripció i els objectius del projecte a realitzar. Presenta un primer anàlisi inicial de requeriments, parla sobre la tecnologia que s'utilitzarà i planifica les tasques i recursos necessaris per poder dur a terme el projecte en la planificació estimada.
- ✓ **Document d'anàlisi i disseny, mbano_AnalisiDisseny.pdf:**
Document que fa un anàlisi més exhaustiu que l'anterior. Explica com volem que sigui la nostre aplicació, com la desenvoluparem i quina arquitectura utilitzarà.
- ✓ **Script de creació de base de dades ScriptBaseDadesHSPA.sql:**
Script que serveix per crear la base de dades Microsoft SQL Server 2008 i que introdueix algunes dades per provar l'aplicació.

- ✓ **Document d'instal·lació de l'aplicació, Instal·lació HotelSpa.pdf:**
Manual d'instal·lació dels productes obtinguts a la fase d'implementació.
Explica, pas a pas, com posar en marxa les aplicacions.
- ✓ **Manual d'usuari de l'aplicació, Manual Usuari Hotel Spa.pdf:**
Manual d'usuari (tant per a usuaris com per a administradors) que serveix per donar una explicació ràpida del funcionament de l'aplicació.
- ✓ **Aplicació d'escriptori WpfHotelSpa:**
Solució de l'aplicació d'escriptori. Aquesta solució inclou la part de presentació, realitzada amb WPF, que serà el que veuran els usuaris.
- ✓ **Servei WCF i aplicació per hostatjar-lo, HotelSpaWCF i Servidor:**
HotelSpaWCF és el servei que permetrà a l'aplicació d'escriptori obtenir i modificar la informació de la base de dades. Servidor és una aplicació de consola encarregada de hostatjar i posar en marxa el servei.
- ✓ **Llibreria de classe LinqHotelSpa:**
Llibreria de classes de Linq to Sql que serveix per obtenir les classes d'entitat de les taules de la base de dades.
- ✓ **Memòria final del projecte, mbano_Memòria.pdf:**
(Aquest document). Document final que fa un resum de tot el treball realitzat per finalitzar el projecte.
- ✓ **Presentació virtual del projecte, mbano_Presentació_HotelSpa.avi:**
Vídeo de presentació del Treball de Fi de Carrera que consta de dues parts. La primera és la presentació del projecte i la segona part es una breu demostració de l'aplicació final.

2. Requeriments inicials

A continuació farem una descripció dels requeriments funcionals i no funcionals del sistema, tenint com escenari de partida la informació facilitada per l'hotel SPA. Els requeriments inicials de l'escenari de partida son els següents:

- ❖ Interfície d'usuari (IU) rica, gràfica, intuïtiva i agradable.
- ❖ La IU ha de tenir un auto aprenentatge ràpid per part de l'usuari.
- ❖ Els usuaris han de poder obtenir tota la informació possible sobre les activitats, recursos i serveis de l'hotel.
- ❖ Els administradors han de poder gestionar els usuaris, les activitats, els recursos i els serveis.

2.1. Requeriments funcionals

Els actors que interactuaran amb el nostre sistema son clarament dos, usuari i administrador. Un administrador és un usuari amb privilegis, que podrà gestionar totes les funcionalitats del programa. Podríem haver diferenciat entre diferents tipus d'administradors(un per a cada àrea: serveis, activitats, recursos), però en aquest cas els usuaris "administradors" tindran accés total a l'aplicació. Els requeriments funcionals que ha d'oferir la nostre aplicació són els següents:

- ❖ **Login:**
 - L'usuari de l'aplicació, per poder accedir al sistema, té que entrar el seu nom d'usuari i contrasenya. Si és un usuari "normal" podrà accedir a la diferent informació que dona l'aplicació i si ho desitja fer peticions, reserves o inscripcions. En el cas de que l'usuari sigui "administrador", (això es controlarà amb un atribut booleà anomenat *Administrador*), accedirà a la gestió de tots els serveis, recursos i activitats de l'hotel o, podrà entrar al sistema com un usuari "normal".
- ❖ **Gestió Usuaris:**
 - Els administradors han de poder afegir nous usuaris, modificar les seves dades així com la seva contrasenya i també poder eliminar/recuperar-los.
 - Un usuari ha de poder modificar les seves pròpies dades encara que no estarà permès que es pugui eliminar a si .
 - No poden existir dos persones amb el mateix nom d'usuari.
- ❖ **Gestió Serveis:**
 - Els administradors han de poder donar d'alta nous serveis de l'hotel així com modificar, eliminar o recuperar els ja existents.
 - Han de poder consultar totes les peticions dels clients.

- Els usuaris han de poder fer peticions al servei d'habitacions, peticions de material, servei de despertador. Poden accedir al sistema i aquest els mostrarà els serveis disponibles mencionats anteriorment, podent fer les peticions que desitgin així com cancel·lar-les.

- ❖ **Gestió Recursos**
 - Els administradors han de poder afegir, modificar, eliminar o recuperar qualsevol recurs de que disposi l'hotel(sales de reunions, pistes d'esport etc.).
 - També podran fer consultes sobre els recursos de que disposa l'hotel així com de les reserves d'aquests recursos.
 - Els usuaris han de poder consultar els recursos que els ofereix l'hotel així com fer una reserva o cancel·lar-la. L'usuari només podrà reservar un recurs sempre que estigui lliure en la data sol·licitada.

- ❖ **Gestió Activitats:**
 - Els administradors ha de poder gestionar les activitats que ofereix l'hotel com ara el gimnàs, massatgista, sala de conferències,etc...
 - Tindran que poder consultar les activitats oferides per l'hotel i les inscripcions realitzades.
 - Els usuaris han de poder consultar les activitats realitzades al recinte del hotel i reservar plaça o cancel·lar la seva inscripció. Si l'usuari vol fer una inscripció de una activitat que té un nombre de places limitades el sistema l'informarà de que no ho pot fer.

- ❖ **Consultes i llistats**
 - Els administradors podran obtenir llistats dels usuaris, activitats, serveis, recursos, inscripcions, peticions i reserves.
 - Els usuaris podran consultar totes les inscripcions, reserves i peticions que hagin realitzat.

2.2. Requeriments no funcionals

Els requeriments no funcionals son:

- ❖ Les interfícies d'usuari es desenvoluparan amb **WPF**.
- ❖ Les comunicacions entre la interfície amb la lògica de negoci i la base de dades es farà amb **WCF**.
- ❖ S'utilitzarà **LINQ-to-SQL** pel tractament de dades entre l'aplicació i la base de dades.

3. Anàlisi del sistema

3.1. Diagrama de casos d'ús

A continuació mostrem alguns dels casos d'ús més importants del sistema. A l'apartat següent farem una especificació textual dels casos d'ús, descompostos en passos numerats, de cada un dels quals indicarem si els fa el sistema o bé un actor i quines dades hi intervenen.

A diferència del document “Anàlisi i Disseny” entregat a la PAC2, s’han detectat uns nous casos d’ús que fan referència a les gestions que poden realitzar els administradors:

Aquest nous casos d’ús són:

- Gestió Usuaris: S’ha afegit el cas d’ús “*Recuperar Usuari*”.
- Gestió Serveis: S’ha afegit el cas d’ús “*Recuperar Servei*”.
- Gestió Recursos: S’ha afegit el cas d’ús “*Recuperar Recurs*”.
- Gestió Activitats: “S’ha afegit el cas d’ús “*Recuperar Activitat*”.

3.1.1. Diagrama de casos d'ús Accedir al Sistema

Quan l’usuari vol accedir al sistema s’executarà el cas d’ús Login, que comprovarà si el nom d’usuari i contrasenya són correctes i, depenent de si és administrador o usuari normal, donarà accés a una part de l’aplicació o a tota la gestió.

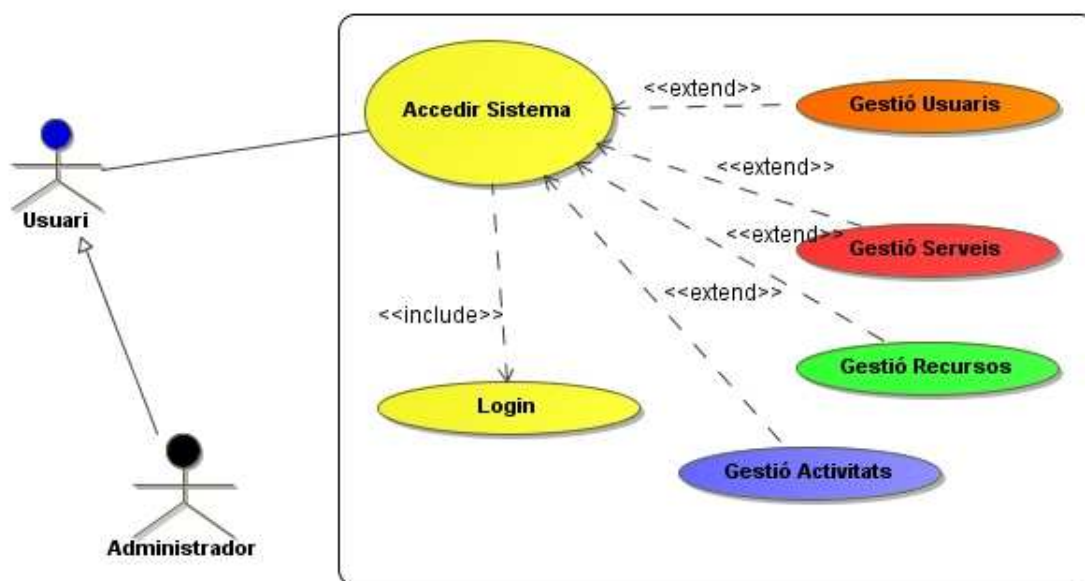


Figura 5. Cas d'ús Accedir Sistema

3.1.2. Diagrama de casos d'ús General

Com hem comentat a l'apartat anterior els actors que interactuaran amb el sistema són l'actor Usuari i l'actor Administrador. L'administrador és una especialització de l'actor usuari perquè a part de poder fer el mateix que un usuari podrà fer altres gestions amb el sistema que només es poden dur a terme si ets un usuari administrador. En el següent diagrama no es reflecteix aquesta especialització per motius de claredat.

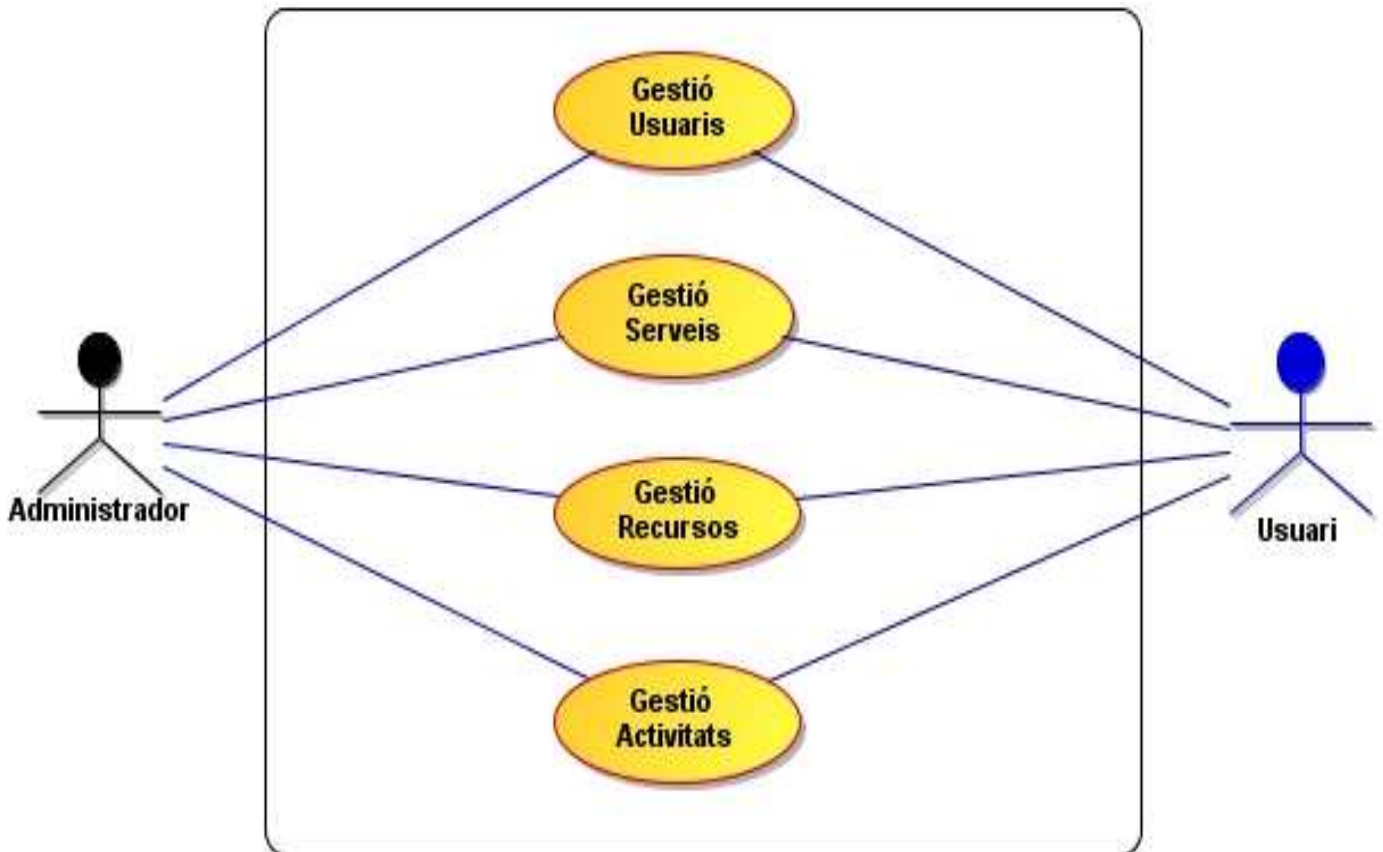


Figura 6. Cas d'ús general del sistema.

3.1.3. Diagrama de casos d'us Gestió Usuaris

En aquest diagrama veiem les funcions que pot realitzar l'administrador sobre els usuaris. Podrà donar d'alta un nou usuari, modificar les dades d'un usuari i també podrà eliminar-lo del sistema.

Al modificar o eliminar un usuari s'engegarà el cas d'ús Cercar Usuari per poder seleccionar l'usuari desitjat i modificar-lo o eliminar-lo.

També pot obtenir un llistat de tots els usuaris del sistema.

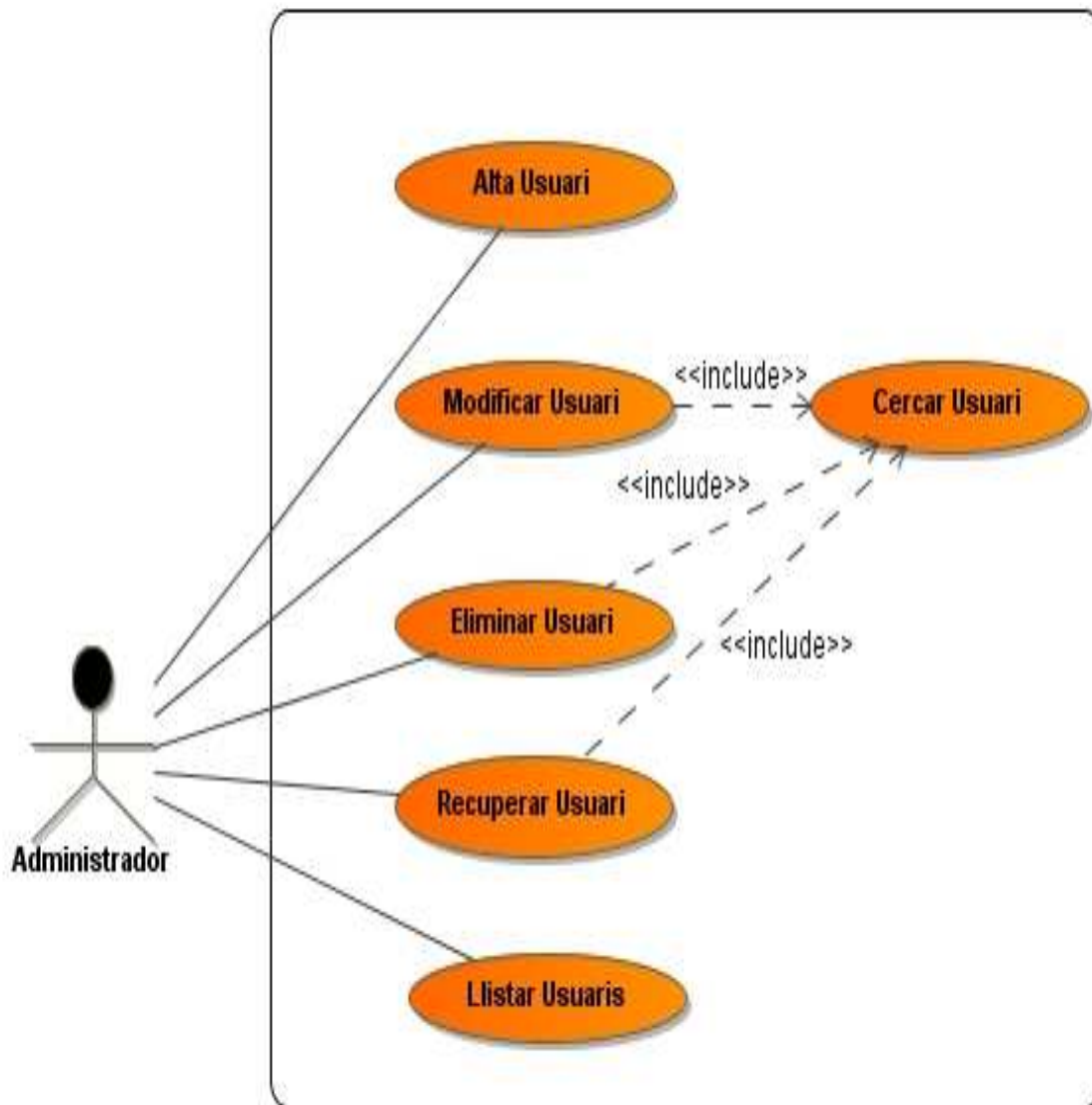


Figura 7. Cas d'ús Gestió Usuaris.

3.1.4. Diagrama de casos d'ús Gestió Serveis

En aquest cas d'ús intervenen els dos actors: Usuari i Administrador.

Els usuaris poden consultar els serveis que els ofereix l'hotel per, si els interessa, realitzar una petició. També poden consultar les seves peticions i, si es necessari, cancel·lar-les.

Els administradors també poden afegir, modificar, eliminar o recuperar serveis, així com obtenir un llistat de les peticions que s'hagin fet. També poden cancel·lar una petició.

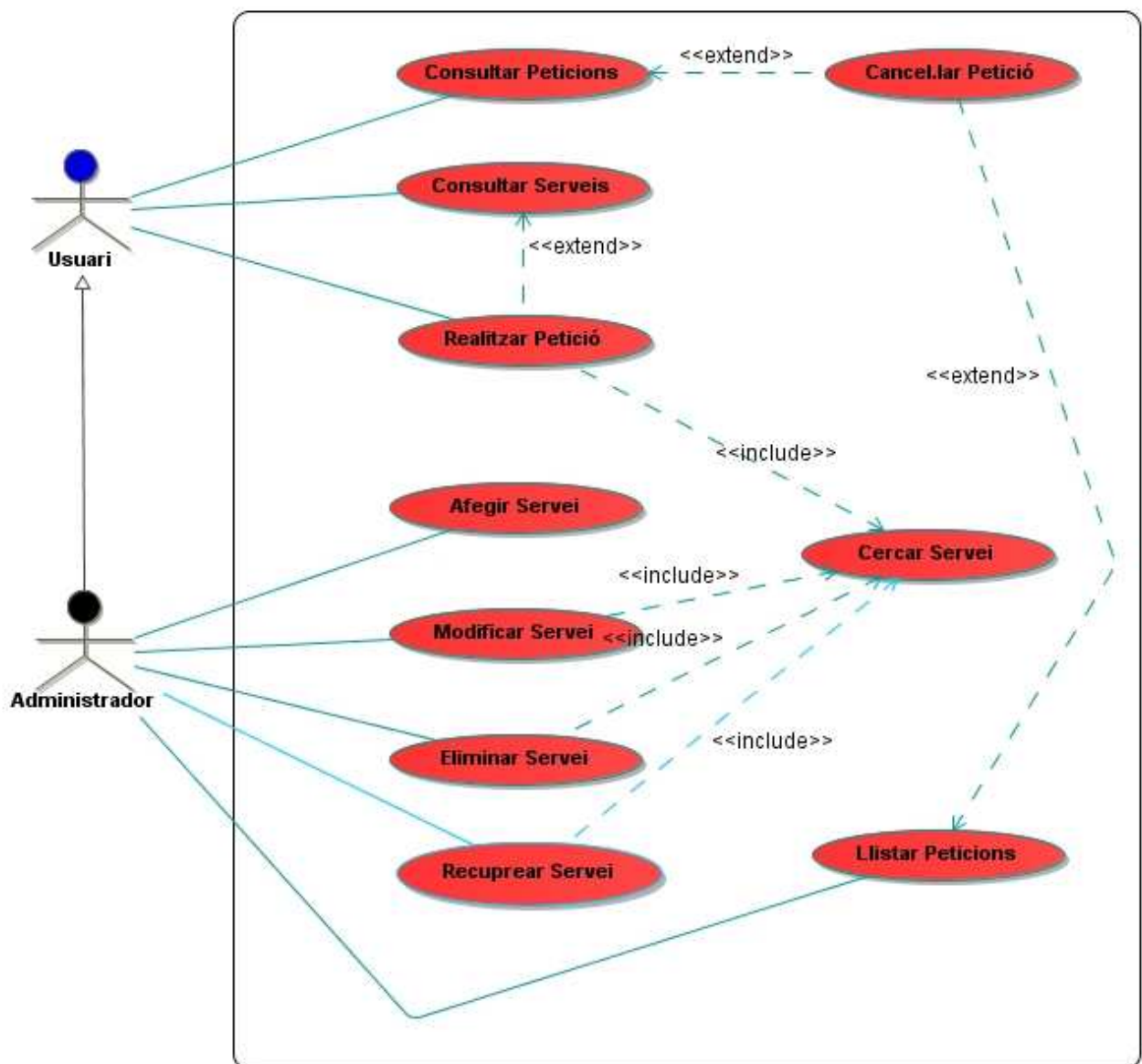


Figura 8. Cas d'ús Gestió Serveis.

3.1.5. Diagrama de casos d'ús Gestió Recursos

Els actors que intervenen són Usuari i Administrador.

En aquest cas d'ús veiem com els administradors gestionaran els recursos de forma molt semblant a les peticions de serveis. Podran donar d'alta, modificar o eliminar un recurs, llistar les reserves realitzades pels clients de l'hotel i també podran cancel·lar-les.

Els usuaris poden consultar els recursos de que disposa l'hotel, reservar-los (si hi han places lliures) o cancel·lar les seves reserves.

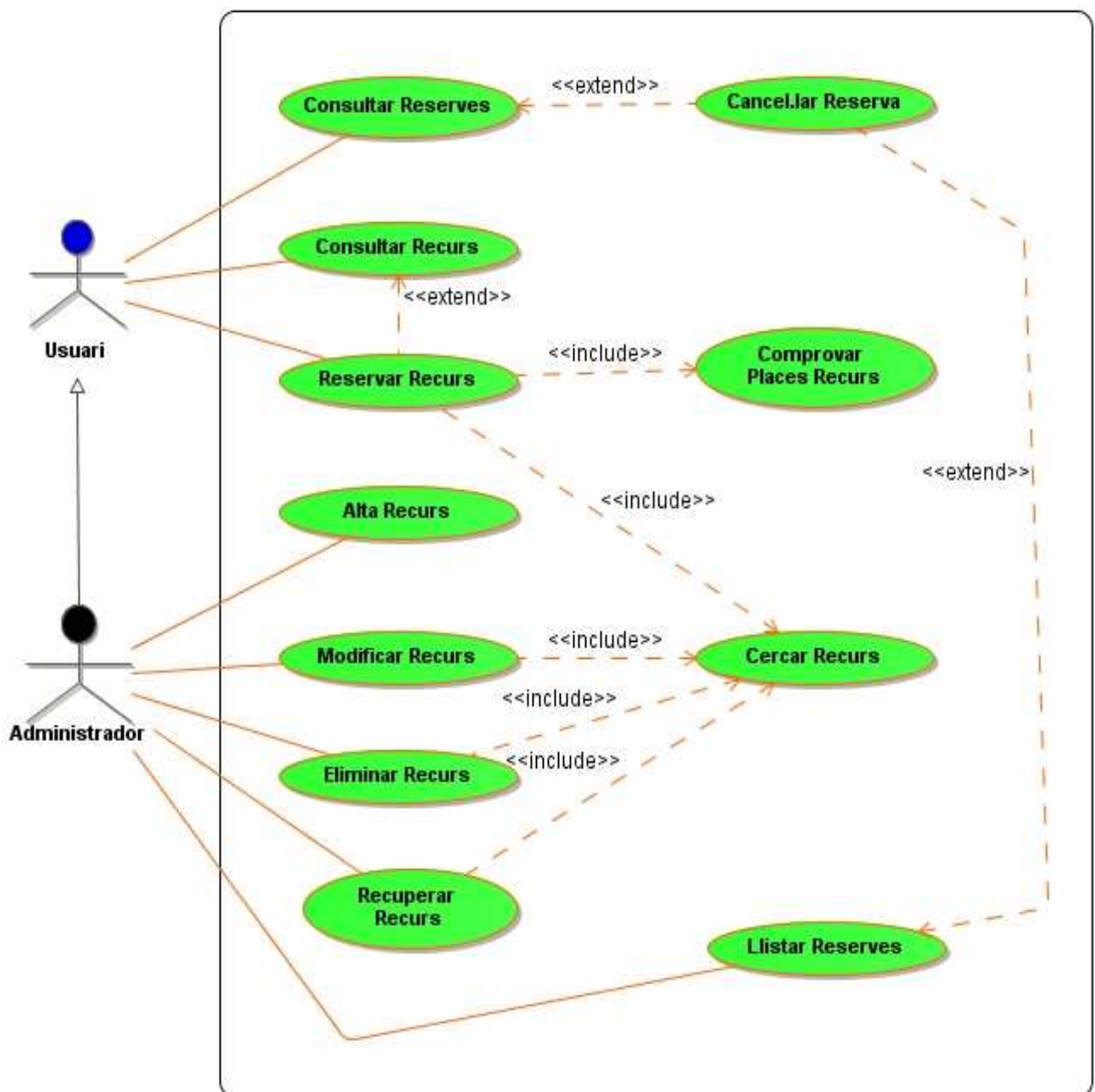


Figura 9. Cas d'ús Gestió Recursos.

3.1.6. Diagrama de casos d'ús Gestió Activitats

Com als casos d'ús anteriors, els actors que intervenen en aquest cas d'ús són Usuari i Administrador.

El funcionament del cas d'ús Gestió Activitats és el mateix que el de Gestió Recursos. Els administrador podran donar d'alta, modificar o eliminar activitats, llistar les inscripcions dels usuaris i cancel·lar-les.

Els usuaris podran consultar les activitats que es planifiquen a l'hotel, inscriure's en les que estiguin interessats o cancel·lar les seves inscripcions. També podran consultar les seves pròpies inscripcions.

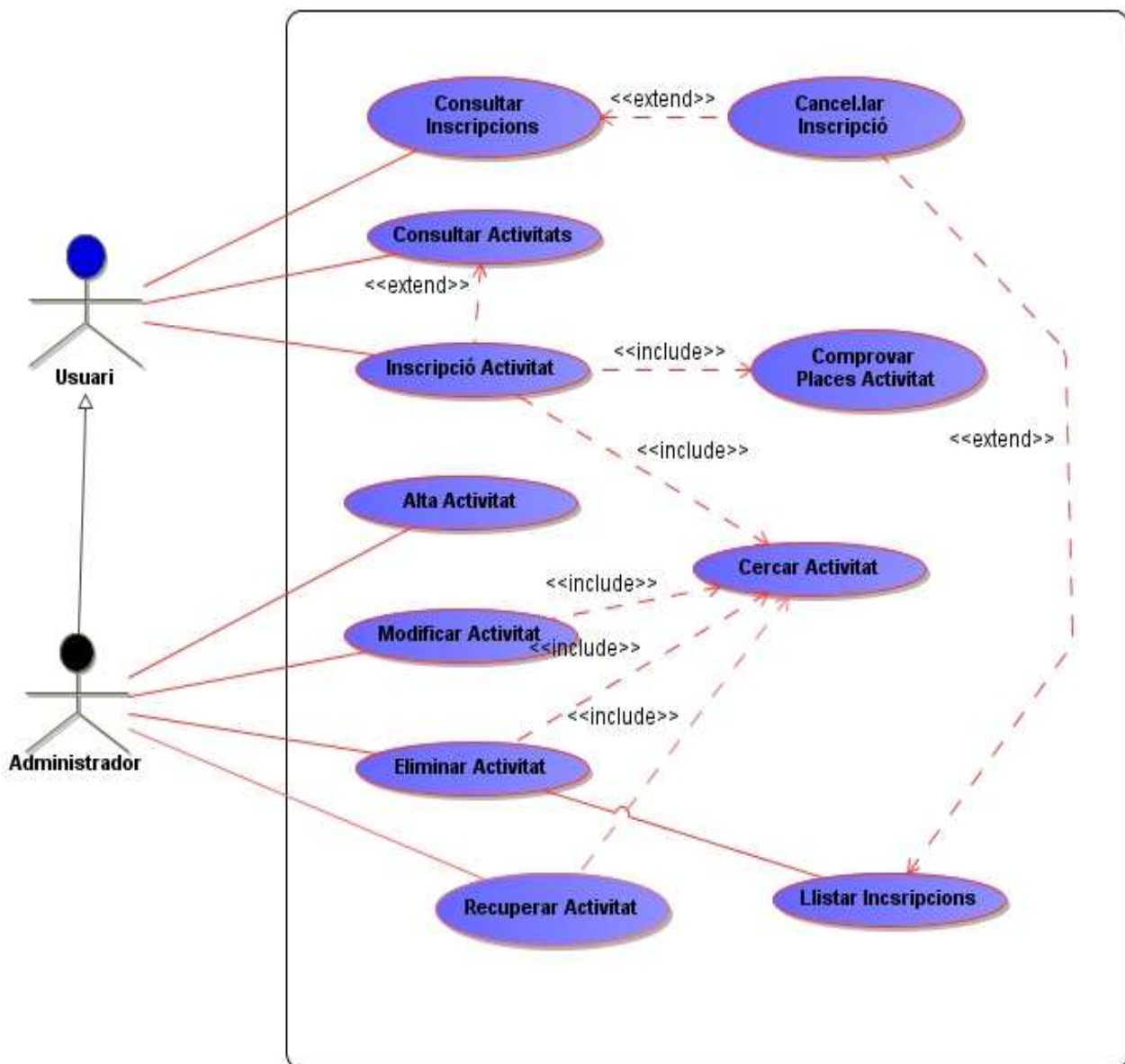


Figura 10. Cas d'ús Gestió Activitats.

3.2. Especificació textual dels casos d'ús anteriors

A continuació farem una especificació textual d'alguns dels casos d'ús més importants.

3.2.1. Casos d'ús Accedir Sistema

Cas d'ús Accedir Sistema	
Resum de la funcionalitat	Un usuari o administrador accedeix al sistema.
Actors	Usuari o Administrador.
Casos d'ús relacionats	Login.
Precondició	Cap.
Postcondició	L'usuari o administrador ha accedit al sistema.
Procés normal principal	<ol style="list-style-type: none"> 1. L'usuari vol accedir al sistema. 2. El sistema engega el cas d'ús Login. 3. El sistema demana el nom d'usuari i contrasenya. 4. L'usuari accedeix al sistema.
Alternatives de procés i excepcions	<ol style="list-style-type: none"> 4a. El nom d'usuari o la contrasenya no són correctes. 4a1. El sistema mostra un missatge d'error i tornem al pas 3.

Cas d'ús Login	
Resum de la funcionalitat	Es comprova el nom d'usuari i la contrasenya introduïts.
Actors	Usuari o Administrador.
Casos d'ús relacionats	Accedir Sistema.
Precondició	Les dades introduïdes són correctes.
Postcondició	L'usuari o administrador ha accedit al sistema.
Procés normal principal	<ol style="list-style-type: none"> 1. El sistema demana el nom d'usuari i la contrasenya. 2. L'usuari o l'administrador introdueixen les dades demanades. 3. El sistema comprova les dades introduïdes a la base de dades. 4. Accedeixen al sistema.
Alternatives de procés i excepcions	<ol style="list-style-type: none"> 4a. El nom d'usuari o la contrasenya no són correctes. 4a1. El sistema mostra un missatge d'error i finalitza el cas d'ús.

3.2.2. Casos d'ús Gestió Usuaris

Cas d'ús Alta Usuari	
Resum de la funcionalitat	Un administrador dona d'alta un nou usuari.
Actors	Administrador.
Casos d'ús relacionats	Cap.
Precondició	L'usuari no existeix.
Postcondició	L'usuari ha sigut donat d'alta al sistema.
Procés normal principal	<ol style="list-style-type: none"> 1. El sistema mostra les opcions de que disposa l'administrador. 2. L'administrador selecciona Gestió Usuaris. 3. L'administrador selecciona l'opció d'afegir un nou usuari. 4. L'administrador introdueix les dades del nou usuari. 5. El sistema afegeix un nou usuari a la base de dades.
Alternatives de procés i excepcions	<ol style="list-style-type: none"> 5a. El nom d'usuari ja existeix a la base de dades i tornem al pas 4.

Cas d'ús Modificar Usuari	
Resum de la funcionalitat	Un administrador modifica alguna dada d'un usuari.
Actors	Administrador.
Casos d'ús relacionats	Cercar Usuari.
Precondició	Cap.
Postcondició	Les dades de l'usuari han estat modificades.
Procés normal principal	<ol style="list-style-type: none"> 1. El sistema mostra les opcions de que disposa l'administrador. 2. L'administrador selecciona Gestió Usuaris. 3. L'administrador selecciona modificar les dades d'un usuari. 4. El sistema engega el cas d'ús <u>Cercar Usuari</u>. 5. El sistema mostra a l'administrador un llistat amb els usuaris de que disposa el sistema. 6. L'administrador en selecciona un. 7. L'administrador modifica les dades de l'usuari seleccionat. 8. El sistema enregistra la modificació de l'usuari seleccionat.

Cas d'ús Eliminar Usuari	
Resum de la funcionalitat	Un administrador elimina un usuari del sistema.
Actors	Administrador.
Casos d'ús relacionats	Cercar Usuari.
Precondició	Cap.
Postcondició	L'usuari ha estat eliminat del sistema.
Procés normal principal	<ol style="list-style-type: none"> 1. El sistema mostra les opcions de que disposa l'administrador. 2. L'administrador selecciona Gestió Usuaris. 3. L'administrador selecciona eliminar un usuari. 4. El sistema engega el cas d'ús <u>Cercar Usuari</u>. 5. El sistema mostra al administrador un llistat amb els usuaris de que disposa el sistema. 6. L'administrador en selecciona un. 7. L'administrador dona l'ordre d'eliminar l'usuari. 8. El sistema elimina l'usuari seleccionat de la base de dades.

Cas d'ús Recuperar Usuari	
Resum de la funcionalitat	Un administrador recupera un usuari eliminat del sistema.
Actors	Administrador.
Casos d'ús relacionats	Cercar Usuari.
Precondició	Cap.
Postcondició	L'usuari ha estat recuperar al sistema.
Procés normal principal	<ol style="list-style-type: none"> 1. El sistema mostra les opcions de que disposa l'administrador. 2. L'administrador selecciona Gestió Usuaris. 3. L'administrador selecciona Recupera Usuari. 4. El sistema engega el cas d'ús <u>Cercar Usuari</u>. 5. El sistema mostra al administrador un llistat amb els usuaris de que disposa el sistema. 6. L'administrador en selecciona un. 7. L'administrador dona l'ordre de recuperar l'usuari. 8. El sistema recupera l'usuari seleccionat de la base de dades.

Cas d'ús Llistar Usuaris	
Resum de la funcionalitat	Un administrador obté el llistat d'usuaris del sistema.
Actors	Administrador.
Casos d'ús relacionats	Cercar Usuari.

Precondició	Cap.
Postcondició	Es mostra un llistat amb els usuaris del sistema.
Procés normal principal	<ol style="list-style-type: none"> 1. El sistema mostra les opcions de que disposa l'administrador. 2. L'administrador selecciona Gestió Usuaris. 3. L'administrador selecciona Llistat Usuaris. 4. El sistema mostra tots els usuaris.
Alternatives de procés i excepcions	<ol style="list-style-type: none"> 4a. L'administrador imprimeix el llistat d'usuaris. 4b. L'administrador finalitza el cas d'ús.

Cas d'ús Cercar Usuaris	
Resum de la funcionalitat	Es cerca un usuari a partir del nom d'usuari o del nom i cognoms.
Actors	Administrador.
Casos d'ús relacionats	Modificar Usuari, Eliminar Usuari.
Precondició	Cap.
Postcondició	S'ha cercat i s'ha trobat un usuari.
Procés normal principal	<ol style="list-style-type: none"> 1. El sistema demana les dades necessàries de l'usuari a cercar. 2. L'administrador les introdueix.. 3. El sistema mostra un llistat amb les coincidències. 4. L'administrador selecciona un usuari de la llista.
Alternatives de procés i excepcions	<ol style="list-style-type: none"> 3a. No s'ha trobat cap resultat. 3a1. El sistema notifica a l'usuari que no s'ha trobat cap coincidència i tornem al pas 1.

3.2.3. Casos d'ús Gestió Serveis

Cas d'ús Alta Servei	
Resum de la funcionalitat	Un administrador dona d'alta un nou servei.
Actors	Administrador.
Casos d'ús relacionats	Cap.
Precondició	El servei no existeix.
Postcondició	El servei ha sigut donat d'alta al sistema.
Procés normal principal	<ol style="list-style-type: none"> 1. El sistema mostra les opcions de que disposa l'administrador. 2.L'administrador selecciona la Gestió Serveis. 3.L'administrador selecciona l'opció d'afegir un nou servei. 4.L'administrador introdueix les dades del nou servei. 5.El sistema afegeix un nou servei a la base de dades.
Alternatives de procés i excepcions	5a.El servei ja existeix a la base de dades i tornem al pas 4.

Cas d'ús Modificar Servei	
Resum de la funcionalitat	Un administrador modifica alguna dada d'un servei.
Actors	Administrador.
Casos d'ús relacionats	Cercar Servei.
Precondició	Cap.
Postcondició	Les dades del servei han estat modificades.
Procés normal principal	<ol style="list-style-type: none"> 1. El sistema mostra les opcions de que disposa l'administrador. 2. L'administrador selecciona Gestió Serveis. 3. L'administrador selecciona modificar les dades d'un servei. 4. El sistema engega el cas d'ús <u>Cercar Servei</u>. 5. El sistema mostra al l'administrador un llistat amb els serveis de que disposa el sistema.

6. L'administrador en selecciona un.
7. L'administrador modifica les dades del servei seleccionat.
8. El sistema enregistra la modificació del servei seleccionat.

Cas d'ús Eliminar Servei	
Resum de la funcionalitat	Un administrador elimina un servei del sistema.
Actors	Administrador.
Casos d'ús relacionats	Cercar Servei.
Precondició	Cap.
Postcondició	El servei ha estat eliminat del sistema.
Procés normal principal	<ol style="list-style-type: none"> 1. El sistema mostra les opcions de que disposa l'administrador. 2. L'administrador selecciona Gestió Serveis. 3. L'administrador selecciona eliminar un servei. 4. El sistema engega el cas d'ús <u>Cercar Servei</u>. 5. El sistema mostra al administrador un llistat amb els serveis de que disposa el sistema. 6. L'administrador en selecciona un. 7. L'administrador dona l'ordre d'eliminar el servei. 8. El sistema elimina el servei seleccionat de la base de dades.

Cas d'ús Recuperar Servei	
Resum de la funcionalitat	Un administrador recupera un servei eliminat del sistema.
Actors	Administrador.
Casos d'ús relacionats	Cercar Servei.
Precondició	Cap.
Postcondició	El servei ha estat recuperar al sistema.
Procés normal principal	<ol style="list-style-type: none"> 1. El sistema mostra les opcions de que disposa l'administrador. 2. L'administrador selecciona Gestió Serveis. 3. L'administrador selecciona Recuperar Servei. 4. El sistema engega el cas d'ús <u>Cercar Servei</u>. 5. El sistema mostra al administrador un llistat amb els serveis de que disposa el sistema. 6. L'administrador en selecciona un. 7. L'administrador dona l'ordre de recuperar el servei. 8. El sistema recupera el servei seleccionat de la base de dades.

Cas d'ús Llistar Peticions	
Resum de la funcionalitat	Un administrador obté el llistat de serveis del sistema.
Actors	Administrador.
Casos d'ús relacionats	Cancel·lar Petició.
Precondició	Cap.
Postcondició	Es mostra un llistat amb els serveis del sistema.
Procés normal principal	<ol style="list-style-type: none"> 1. El sistema mostra les opcions de que disposa l'administrador. 2. L'administrador selecciona Gestió Serveis. 3. L'administrador selecciona llistar peticions. 4. El sistema mostra totes les peticions.
Alternatives de procés i excepcions	<ol style="list-style-type: none"> 5a. L'administrador imprimeix el llistat de peticions 5b. L'administrador finalitza el cas d'ús. 5c. L'administrador selecciona una petició per eliminar-la. 5c1. El sistema engega el cas d'ús <u>Cancel·lar Petició</u>.

Cas d'ús Cercar Servei	
Resum de la funcionalitat	Es cerca un servei a partir del nom del servei o del seu codi.

Actors	Usuari o Administrador.
Casos d'ús relacionats	Modificar Servei, Eliminar Servei.
Precondició	Cap.
Postcondició	S'ha cercat i s'ha trobat un servei.
Procés normal principal	<ol style="list-style-type: none"> 1. El sistema demana les dades necessàries del servei a cercar. 2. L'administrador o l'usuari les introdueix.. 3. El sistema mostra un llistat amb les coincidències. 4. L'administrador o l'usuari selecciona un servei de la llista.
Alternatives de procés i excepcions	<p>3a. No s'ha trobat cap resultat.</p> <p>3a1. El sistema notifica a l'usuari que no s'ha trobat cap coincidència i tornem al pas 1.</p>

Cas d'ús Consultar Serveis	
Resum de la funcionalitat	L'usuari consulta els serveis que ofereix l'hotel.
Actors	Usuari.
Casos d'ús relacionats	Realitzar Petició.
Precondició	Cap
Postcondició	S'obté la informació de tots els serveis de l'hotel.
Procés normal principal	<ol style="list-style-type: none"> 1. El sistema mostra les opcions de que disposa l'usuari. 2. L'usuari selecciona Gestió Serveis. 3. L'usuari selecciona Consultar Serveis. 4. El sistema mostra tots els serveis de que disposa l'hotel.
Alternatives de procés i excepcions	<p>4a. No s'ha trobat cap resultat.</p> <p>4a1. El sistema notifica a l'usuari que no s'ha trobat cap coincidència i finalitza el cas d'ús.</p> <p>5a. L'usuari vol Realitzar Petició</p> <p>5a1. El sistema engega el cas d'ús <u>Realitzar Petició</u>.</p>

Cas d'ús Realitzar Petició	
Resum de la funcionalitat	L'usuari realitza una petició d'un servei de l'hotel.
Actors	Usuari.
Casos d'ús relacionats	Cercar Servei, Consultar Serveis.
Precondició	Cap.
Postcondició	S'ha realitzat una petició d'un servei.
Procés normal principal	<ol style="list-style-type: none"> 1. El sistema mostra les opcions de que disposa l'usuari. 2. L'usuari selecciona Gestió Serveis. 3. L'usuari selecciona Realitzar Petició. 4. El sistema engega el cas d'ús <u>Cercar Servei</u>. 5. El sistema mostra a l'usuari un llistat amb els serveis de que disposa el sistema. 6. L'usuari en selecciona un. 7. L'usuari dona l'ordre de realitzar la petició del servei seleccionat. 8. El sistema enregistra la petició realitzada a la base de dades.
Alternatives de procés i excepcions	<p>5a. No s'ha trobat cap resultat.</p> <p>5a1. El sistema notifica a l'usuari que no s'ha trobat cap coincidència i finalitza el cas d'ús.</p>

Cas d'ús Consultar Peticions	
Resum de la funcionalitat	L'usuari consulta les peticions realitzades.
Actors	Usuari.
Casos d'ús relacionats	Cancel·lar Petició.
Precondició	L'usuari ha de tenir peticions realitzades.
Postcondició	S'obtenen totes les peticions de l'usuari.

Procés normal principal	<ol style="list-style-type: none"> 1. El sistema mostra les opcions de que disposa l'usuari. 2. L'usuari selecciona Gestió Serveis. 3. L'usuari selecciona Consultar Peticions. 4. El sistema mostra totes les peticions realitzades per aquest usuari.
Alternatives de procés i excepcions	<ol style="list-style-type: none"> 5a. No s'ha trobat cap resultat. 5a1. El sistema notifica a l'usuari que no s'ha trobat cap coincidència i finalitza el cas d'ús. 5b. L'usuari vol cancel·lar una petició. 5b1. El sistema engega el cas d'ús <u>Cancel·lar Petició</u>.

Cas d'ús Cancel·lar Petició	
Resum de la funcionalitat	L'usuari fa una cancel·lació d'una petició.
Actors	Usuari o Administrador.
Casos d'ús relacionats	Consultar Peticions, Llistar Peticions.
Precondició	Cap.
Postcondició	S'ha cancel·lat una petició.
Procés normal principal	<ol style="list-style-type: none"> 1a. L'usuari vol cancel·lar una petició i des del cas d'ús Consultar Peticions indica que en vol eliminar una. 2a. L'usuari selecciona la petició que vol cancel·lar. 1b. L'administrador vol cancel·lar una petició des del cas d'ús Llistar Peticions. 2b. Selecciona la petició a cancel·lar. 3. El sistema elimina la petició seleccionada de la base de dades.
Alternatives de procés i excepcions	<ol style="list-style-type: none"> 3a. La petició a cancel·lar és d'una data anterior a l'actual. 3a1. El Sistema informa de que no es pot cancel·lar una petició d'un servei ja realitzat i finalitza el cas d'ús..

3.2.4. Casos d'ús Gestió Recursos

Cas d'ús Alta Recurs	
Resum de la funcionalitat	Un administrador dona d'alta un nou recurs.
Actors	Administrador.
Casos d'ús relacionats	Cap.
Precondició	El recurs no existeix.
Postcondició	El recurs ha sigut donat d'alta al sistema.
Procés normal principal	<ol style="list-style-type: none"> 1. El sistema mostra les opcions de que disposa l'administrador. 2.L'administrador selecciona Gestió Recursos. 3.L'administrador selecciona l'opció d'afegir un nou recurs. 4.L'administrador introdueix les dades del nou recurs. 5.El sistema afegeix un nou recurs a la base de dades.
Alternatives de procés i excepcions	5a.El recurs ja existeix a la base de dades i tornem al pas 4.

Cas d'ús Modificar Recurs	
Resum de la funcionalitat	Un administrador modifica alguna dada d'un recurs.
Actors	Administrador.
Casos d'ús relacionats	Cercar Recurs.
Precondició	Cap.
Postcondició	Les dades del recurs han estat modificades.
Procés normal principal	1. El sistema mostra les opcions de que disposa l'administrador.

	<ol style="list-style-type: none"> 2. L'administrador selecciona Gestió Recursos. 3. L'administrador selecciona modificar les dades d'un recurs. 4. El sistema engega el cas d'ús <u>Cercar Recurs</u>. 5. El sistema mostra a l'administrador un llistat amb els recursos de que disposa el sistema. 6. L'administrador en selecciona un. 7. L'administrador modifica les dades del recurs seleccionat. 8. El sistema enregistra la modificació del recurs seleccionat.
--	---

Cas d'ús Eliminar Recurs	
Resum de la funcionalitat	Un administrador elimina un recurs del sistema.
Actors	Administrador.
Casos d'ús relacionats	Cercar Recurs.
Precondició	Cap.
Postcondició	El recurs ha estat eliminat del sistema.
Procés normal principal	<ol style="list-style-type: none"> 1. El sistema mostra les opcions de que disposa l'administrador. 2. L'administrador selecciona Gestió Recursos. 3. L'administrador selecciona eliminar un recurs. 4. El sistema engega el cas d'ús <u>Cercar Recurs</u>. 5. El sistema mostra a l'administrador un llistat amb els recursos de que disposa el sistema. 6. L'administrador en selecciona un. 7. L'administrador dona l'ordre d'eliminar el recurs. 8. El sistema elimina el recurs seleccionat de la base de dades.

Cas d'ús Recuperar Recurs	
Resum de la funcionalitat	Un administrador recupera un recurs del sistema.
Actors	Administrador.
Casos d'ús relacionats	Cercar Recurs.
Precondició	Cap.
Postcondició	El recurs ha estat recuperat del sistema.
Procés normal principal	<ol style="list-style-type: none"> 1. El sistema mostra les opcions de que disposa l'administrador. 2. L'administrador selecciona Gestió Recursos. 3. L'administrador selecciona Recuperar Recurs. 4. El sistema engega el cas d'ús <u>Cercar Recurs</u>. 5. El sistema mostra a l'administrador un llistat amb els recursos de que disposa el sistema. 6. L'administrador en selecciona un. 7. L'administrador dona l'ordre de recuperar el recurs. 8. El sistema recupera el recurs seleccionat de la base de dades.

Cas d'ús Llistar Reserves	
Resum de la funcionalitat	Un administrador obté el llistat de reserves del sistema.
Actors	Administrador
Casos d'ús relacionats	Cancel·lar Reserva.
Precondició	Cap.
Postcondició	Es mostra un llistat amb les reserves del sistema.
Procés normal principal	<ol style="list-style-type: none"> 1. El sistema mostra les opcions de que disposa l'administrador. 2. L'administrador selecciona Gestió Recursos. 3. L'administrador selecciona Llistat Reserves. 4. El sistema mostra totes les reserves.
Alternatives de procés i excepcions	<ol style="list-style-type: none"> 4a. L'administrador imprimeix el llistat de reserves. 4b. L'administrador finalitza el cas d'ús. 4c. L'administrador selecciona una reserva per eliminar-la.

4c1. El sistema engega el cas d'ús Cancel·lar Reserva.

Cas d'ús Cercar Recurs	
Resum de la funcionalitat	Es cerca un recurs a partir del nom del recurs o del seu codi.
Actors	Usuari o Administrador.
Casos d'ús relacionats	Modificar Recurs, Eliminar Recurs.
Precondició	Cap.
Postcondició	S'ha cercat i s'ha trobat un recurs.
Procés normal principal	<ol style="list-style-type: none"> 1. El sistema demana les dades necessàries del recurs a cercar. 2. L'administrador o l'usuari les introdueix.. 3. El sistema mostra un llistat amb les coincidències. 4. L'administrador o l'usuari selecciona un recurs de la llista.
Alternatives de procés i excepcions	3a. No s'ha trobat cap resultat. 3a1. El sistema notifica a l'usuari que no s'ha trobat cap coincidència i tornem al pas 1.

Cas d'ús Consultar Recurs	
Resum de la funcionalitat	L'usuari consulta els recursos que ofereix l'hotel.
Actors	Usuari.
Casos d'ús relacionats	Realitzar Petició.
Precondició	Cap.
Postcondició	S'obté la informació de tots els recursos de l'hotel.
Procés normal principal	<ol style="list-style-type: none"> 1. El sistema mostra les opcions de que disposa l'usuari. 2. L'usuari selecciona Gestió Recursos. 3. L'usuari selecciona Consultar Recurs. 4. El sistema mostra tots els recursos de que disposa l'hotel.
Alternatives de procés i excepcions	4a. No s'ha trobat cap resultat. 4a1. El sistema notifica a l'usuari que no s'ha trobat cap coincidència i finalitza el cas d'ús. 4b. L'usuari vol fer una reserva. 4b1. El sistema engega el cas d'ús <u>Reservar Recurs</u> .

Cas d'ús Reservar Recurs	
Resum de la funcionalitat	L'usuari realitza una reserva d'un recurs de l'hotel.
Actors	Usuari.
Casos d'ús relacionats	Cercar Recurs, Consultar Recurs, Comprovar Places Recurs.
Precondició	Cap.
Postcondició	S'ha realitzat una reserva d'un recurs.
Procés normal principal	<ol style="list-style-type: none"> 1. El sistema mostra les opcions de que disposa l'usuari. 2. L'usuari selecciona Gestió Recursos. 3. L'usuari selecciona Reservar Recurs. 4. El sistema engega el cas d'ús <u>Cercar Recurs</u>. 5. El sistema mostra a l'usuari un llistat amb els recursos de que disposa el sistema. 6. L'usuari en selecciona un. 7. L'usuari dona l'ordre de realitzar la petició del recurs seleccionat. 8. El sistema engega el cas d'ús Comprovar Places Recurs. 9. L'usuari finalitza la reserva. 10. El sistema enregistra la petició realitzada a la base de dades.
Alternatives de procés i excepcions	5a. No s'ha trobat cap resultat. 5a1. El sistema notifica a l'usuari que no s'ha trobat cap coincidència i tornem al pas 4. 8a. No queden places lliures. 8a1. El sistema informa de que no queden places i tornem al pas 4.

Cas d'ús Consultar Reserves	
Resum de la funcionalitat	L'usuari consulta les reserves realitzades.
Actors	Usuari.
Casos d'ús relacionats	Cancel·lar Reserva.
Precondició	L'usuari ha de tenir reserves realitzades.
Postcondició	S'obtenen totes les reserves de l'usuari.
Procés normal principal	<ol style="list-style-type: none"> 1. El sistema mostra les opcions de que disposa l'usuari. 2. L'usuari selecciona Gestió Recursos. 3. L'usuari selecciona Consultar Reserves. 4. El sistema mostra totes les reserves realitzades per aquest usuari.
Alternatives de procés i excepcions	<ol style="list-style-type: none"> 4a. No s'ha trobat cap resultat. 4a1. El sistema notifica a l'usuari que no s'ha trobat cap coincidència i finalitza el cas d'ús. 4b. L'usuari vol cancel·lar una reserva. 4b1. El sistema engega el cas d'ús <u>Cancel·lar Reserva</u>.

Cas d'ús Cancel·lar Reserva	
Resum de la funcionalitat	L'usuari fa una cancel·lació d'una reserva.
Actors	Usuari o Administrador.
Casos d'ús relacionats	Consultar Reserves, Llistar Reserves.
Precondició	Cap.
Postcondició	S'ha cancel·lat una reserva.
Procés normal principal	<ol style="list-style-type: none"> 1a. L'usuari vol cancel·lar una reserva d'un recurs i, a partir del cas d'ús <u>Consultar Reserves</u>, indica que en vol eliminar una. 2a. L'usuari selecciona la reserva que vol cancel·lar. 1b. L'administrador vol cancel·lar una reserva a partir del cas d'ús Llistar Reserves. 2b. Selecciona la reserva a cancel·lar. 3. El sistema elimina la reserva seleccionada de la base de dades.
Alternatives de procés i excepcions	<ol style="list-style-type: none"> 3a. La reserva a cancel·lar és d'una data anterior a l'actual. 3a1. El Sistema informa de que no es pot cancel·lar una reserva d'un recurs ja realitzat i finalitza el cas d'ús..

3.2.5. Casos d'ús Gestió Activitats

Cas d'ús Alta Activitat	
Resum de la funcionalitat	Un administrador dona d'alta una nova activitat.
Actors	Administrador.
Casos d'ús relacionats	Cap.
Precondició	L'activitat no existeix.
Postcondició	L'activitat ha sigut donat d'alta al sistema.
Procés normal principal	<ol style="list-style-type: none"> 1. El sistema mostra les opcions de que disposa l'administrador. 2.L'administrador selecciona Gestió Activitats. 3.L'administrador selecciona l'opció d'afegir una nova activitat. 4.L'administrador introdueix les dades de la nova activitat. 5.El sistema afegeix una nova activitat a la base de dades.
Alternatives de procés i excepcions	5a.L'activitat ja existeix a la base de dades i tornem al pas 4.

Cas d'ús Modificar Activitat	
Resum de la funcionalitat	L' administrador modifica alguna dada d'una activitat.
Actors	Administrador.
Casos d'ús relacionats	Cercar Activitat.
Precondició	Cap.
Postcondició	Les dades de l'activitat han estat modificades.
Procés normal principal	<ol style="list-style-type: none"> 1. El sistema mostra les opcions de que disposa l'administrador. 2. L'administrador selecciona Gestió Activitats. 3. L'administrador selecciona modificar les dades d'una activitat. 4. El sistema engega el cas d'ús <u>Cercar Activitat</u>. 5. El sistema mostra a l'administrador un llistat amb les activitats de que disposa el sistema. 6. L'administrador en selecciona una. 7. L'administrador modifica les dades de l'activitat seleccionada. 8. El sistema enregistra la modificació de l'activitat seleccionada.

Cas d'ús Eliminar Activitat	
Resum de la funcionalitat	Un administrador elimina una activitat del sistema.
Actors	Administrador.
Casos d'ús relacionats	Cercar Activitat.
Precondició	Cap.
Postcondició	L'activitat ha estat eliminada del sistema.
Procés normal principal	<ol style="list-style-type: none"> 1. El sistema mostra les opcions de que disposa l'administrador. 2. L'administrador selecciona Gestió Activitats. 3. L'administrador selecciona eliminar una activitat. 4. El sistema engega el cas d'ús <u>Cercar Activitat</u>. 5. El sistema mostra a l'administrador un llistat amb les activitats de que disposa el sistema. 6. L'administrador en selecciona una. 7. L'administrador dona l'ordre d'eliminar l'activitat. 8. El sistema elimina l'activitat seleccionada de la base de dades.

Cas d'ús Recuperar Activitat	
Resum de la funcionalitat	Un administrador recupera una activitat del sistema.
Actors	Administrador.
Casos d'ús relacionats	Cercar Activitat.
Precondició	Cap.
Postcondició	L'activitat ha estat recuperada del sistema.
Procés normal principal	<ol style="list-style-type: none"> 1. El sistema mostra les opcions de que disposa l'administrador. 2. L'administrador selecciona Gestió Activitats. 3. L'administrador selecciona Recuperar Activitat. 4. El sistema engega el cas d'ús <u>Cercar Activitat</u>. 5. El sistema mostra a l'administrador un llistat amb les activitats de que disposa el sistema. 6. L'administrador en selecciona una. 7. L'administrador dona l'ordre de recuperar l'activitat. 8. El sistema recupera l'activitat seleccionada de la base de dades.

Cas d'ús Llistar Inscripcions	
Resum de la funcionalitat	Un administrador obté el llistat d'inscripcions del sistema.
Actors	Administrador
Casos d'ús relacionats	Cancel·lar Inscripció.
Precondició	Cap.
Postcondició	Es mostra un llistat amb les inscripcions del sistema.

Procés normal principal	<ol style="list-style-type: none"> 1. El sistema mostra les opcions de que disposa l'administrador. 2. L'administrador selecciona Gestió Activitats. 3. L'administrador selecciona Llistat Inscripcions. 4. El sistema mostra totes les inscripcions.
Alternatives de procés i excepcions	<ol style="list-style-type: none"> 4a. L'administrador imprimeix el llistat d'inscripcions. 4b. L'administrador finalitza el cas d'ús. 4c. L'administrador selecciona una inscripció per eliminar-la. 4c1. El sistema engega el cas d'ús <u>Cancel·lar Inscripció</u>.

Cas d'ús Cercar Activitat	
Resum de la funcionalitat	Es cerca una activitat a partir del nom de l'activitat o del seu codi.
Actors	Usuari o Administrador.
Casos d'ús relacionats	Modificar Activitat, Eliminar Activitat.
Precondició	Cap.
Postcondició	S'ha cercat i s'ha trobat una activitat.
Procés normal principal	<ol style="list-style-type: none"> 1. El sistema demana les dades necessàries de l'activitat a cercar. 2. L'administrador o l'usuari les introdueix.. 3. El sistema mostra un llistat amb les coincidències. 4. L'administrador o l'usuari selecciona una activitat de la llista.
Alternatives de procés i excepcions	<ol style="list-style-type: none"> 3a. No s'ha trobat cap resultat. 3a1. El sistema notifica a l'usuari que no s'ha trobat cap coincidència i tornem al pas 1.

Cas d'ús Consultar Activitat	
Resum de la funcionalitat	L'usuari consulta les activitats que ofereix l'hotel.
Actors	Usuari.
Casos d'ús relacionats	Realitzar Inscripció.
Precondició	Cap.
Postcondició	S'obté la informació de totes les activitats de l'hotel.
Procés normal principal	<ol style="list-style-type: none"> 1. El sistema mostra les opcions de que disposa l'usuari. 2. L'usuari selecciona Gestió Activitats. 3. L'usuari selecciona Consultar Activitat. 4. El sistema mostra totes les activitats de que disposa l'hotel.
Alternatives de procés i excepcions	<ol style="list-style-type: none"> 4a. No s'ha trobat cap resultat. 4a1. El sistema notifica a l'usuari que no s'ha trobat cap coincidència i finalitza el cas d'ús. 4b. L'usuari vol fer una inscripció. 4b1. El sistema engega el cas d'ús <u>Inscripció Activitat</u>.

Cas d'ús Inscripció Activitat	
Resum de la funcionalitat	L'usuari realitza una inscripció d'una activitat de l'hotel.
Actors	Usuari.
Casos d'ús relacionats	Cercar Activitat, Consultar Activitat, Comprovar Places Activitat.
Precondició	Cap.
Postcondició	S'ha realitzat una inscripció a una activitat.
Procés normal principal	<ol style="list-style-type: none"> 1. El sistema mostra les opcions de que disposa l'usuari. 2. L'usuari selecciona Gestió Activitats. 3. L'usuari selecciona Inscripció Activitat. 4. El sistema engega el cas d'ús <u>Cercar Activitat</u>. 5. El sistema mostra a l'usuari un llistat amb les activitats de que disposa el sistema. 6. L'usuari en selecciona una. 7. L'usuari dona l'ordre de realitzar la inscripció de l'activitat seleccionada.

	<p>8. El sistema engega el cas d'ús Comprovar Places Activitat.</p> <p>9. L'usuari finalitza la inscripció.</p> <p>10. El sistema enregistra la inscripció realitzada a la base de dades.</p>
Alternatives de procés i excepcions	<p>5a. No s'ha trobat cap resultat.</p> <p>5a1. El sistema notifica a l'usuari que no s'ha trobat cap coincidència i tornem al pas 4.</p> <p>8a. No queden places lliures.</p> <p>8a1. El sistema informa de que no queden places i tornem al pas 4.</p>

Cas d'ús Consultar Inscripcions	
Resum de la funcionalitat	L'usuari consulta les inscripcions realitzades.
Actors	Usuari.
Casos d'ús relacionats	Cancel·lar Inscripció.
Precondició	L'usuari ha de tenir inscripcions realitzades.
Postcondició	S'obtenen totes les inscripcions de l'usuari.
Procés normal principal	<p>1. El sistema mostra les opcions de que disposa l'usuari.</p> <p>2. L'usuari selecciona Gestió Activitats.</p> <p>3. L'usuari selecciona Consultar Inscripcions.</p> <p>4. El sistema mostra totes les inscripcions realitzades per aquest usuari.</p>
Alternatives de procés i excepcions	<p>4a. No s'ha trobat cap resultat.</p> <p>4a1. El sistema notifica a l'usuari que no s'ha trobat cap coincidència i finalitza el cas d'ús.</p> <p>4b. L'usuari vol cancel·lar una inscripció.</p> <p>4b1. El sistema engega el cas d'ús <u>Cancel·lar Inscripció</u>.</p>

Cas d'ús Cancel·lar Inscripció	
Resum de la funcionalitat	L'usuari fa una cancel·lació d'una inscripció.
Actors	Usuari o Administrador.
Casos d'ús relacionats	Consultar Inscripcions, Llistar Inscripcions.
Precondició	Cap.
Postcondició	S'ha cancel·lat una inscripció.
Procés normal principal	<p>1a. L'usuari vol cancel·lar una inscripció d'una activitat i, a partir del cas d'ús <u>Consultar Inscripcions</u>, indica que en vol eliminar una.</p> <p>2a. L'usuari selecciona la inscripció que vol cancel·lar.</p> <p>1b. L'administrador vol cancel·lar una inscripció a partir del cas d'ús Llistar Inscripcions.</p> <p>2b. Selecciona la inscripció a cancel·lar.</p> <p>3. El sistema elimina la inscripció seleccionada de la base de dades.</p>
Alternatives de procés i excepcions	<p>3a. La inscripció a cancel·lar és d'una data anterior a l'actual.</p> <p>3a1. El Sistema informa de que no es pot cancel·lar una inscripció d'un activitat ja realitzada i finalitza el cas d'ús..</p>

3.3. Model Conceptual

A continuació veiem les relacions del model conceptual entre les classes d'entitat detectades:

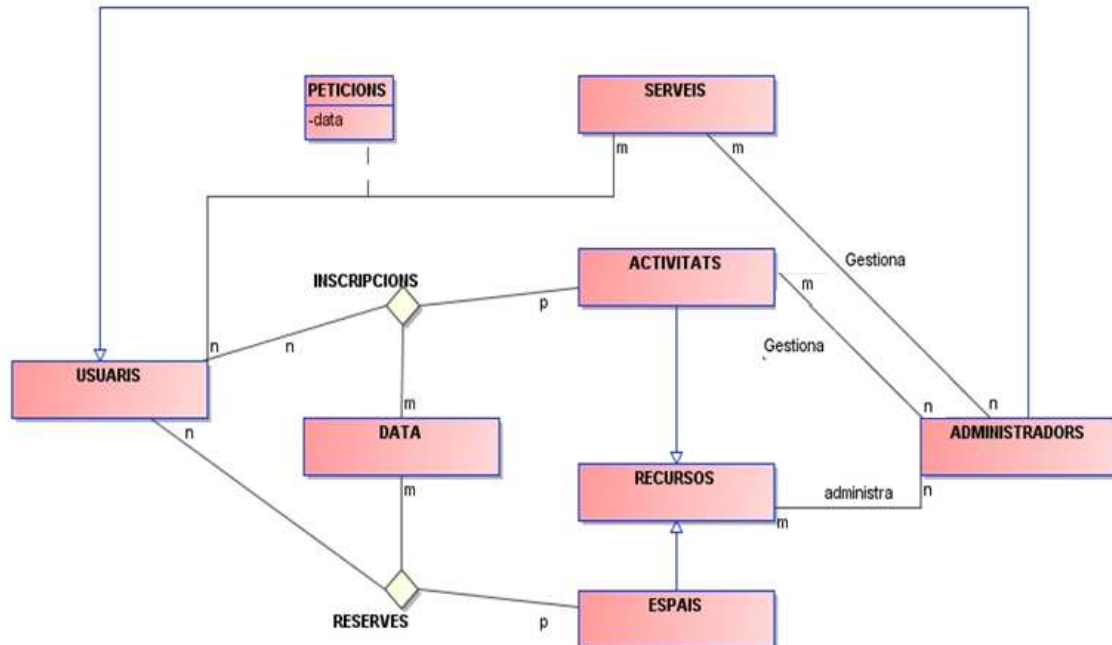


Figura 11. Classes d'entitat Model Conceptual.

3.4. Diagrama de classes d'entitat

3.4.1. Reificació de les classes associatives binàries i de les associacions n-àries

El concepte de classe associativa no existeix als llenguatges de programació i, per tant, es tenen que transformar en classes "normals". Aquest procés s'anomena *reificació*, i el que es fa és substituir la classe associativa per una nova classe que tindrà els mateixos atributs. Aquesta nova classe la relacionarem amb les dues classes que participaven a l'associació original. Per exemple, la classe associativa *Peticions* que hi ha entre *Usuaris* i *Serveis* quedarà :

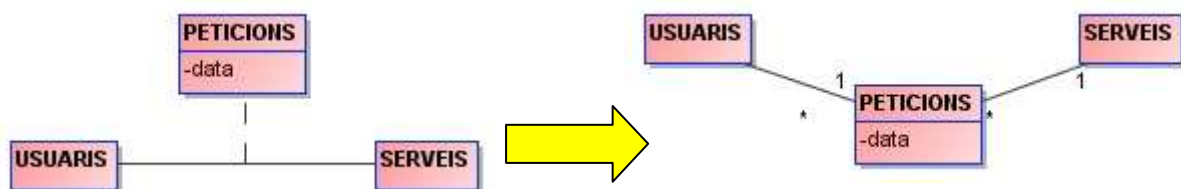


Figura 12. Reificació de classe associativa.

Seguirem el mateix procediment amb les associacions n-àries que trobem.

3.4.2. Altres transformacions

Les entitats Administradors i Usuaris les reduïrem a només una; Usuaris. Per diferenciar si un usuari és un *administrador* o un *usuari normal* afegirem un atribut anomenat “administrador” de tipus booleà a la entitat Usuaris..

Eliminarem l’entitat Recurs i deixarem les seves subclasses. L’entitat Espai la passarem a anomenar Recurs per claredat.

Per últim. substituïm les associacions per atributs.

3.4.3. Diagrama de Classes

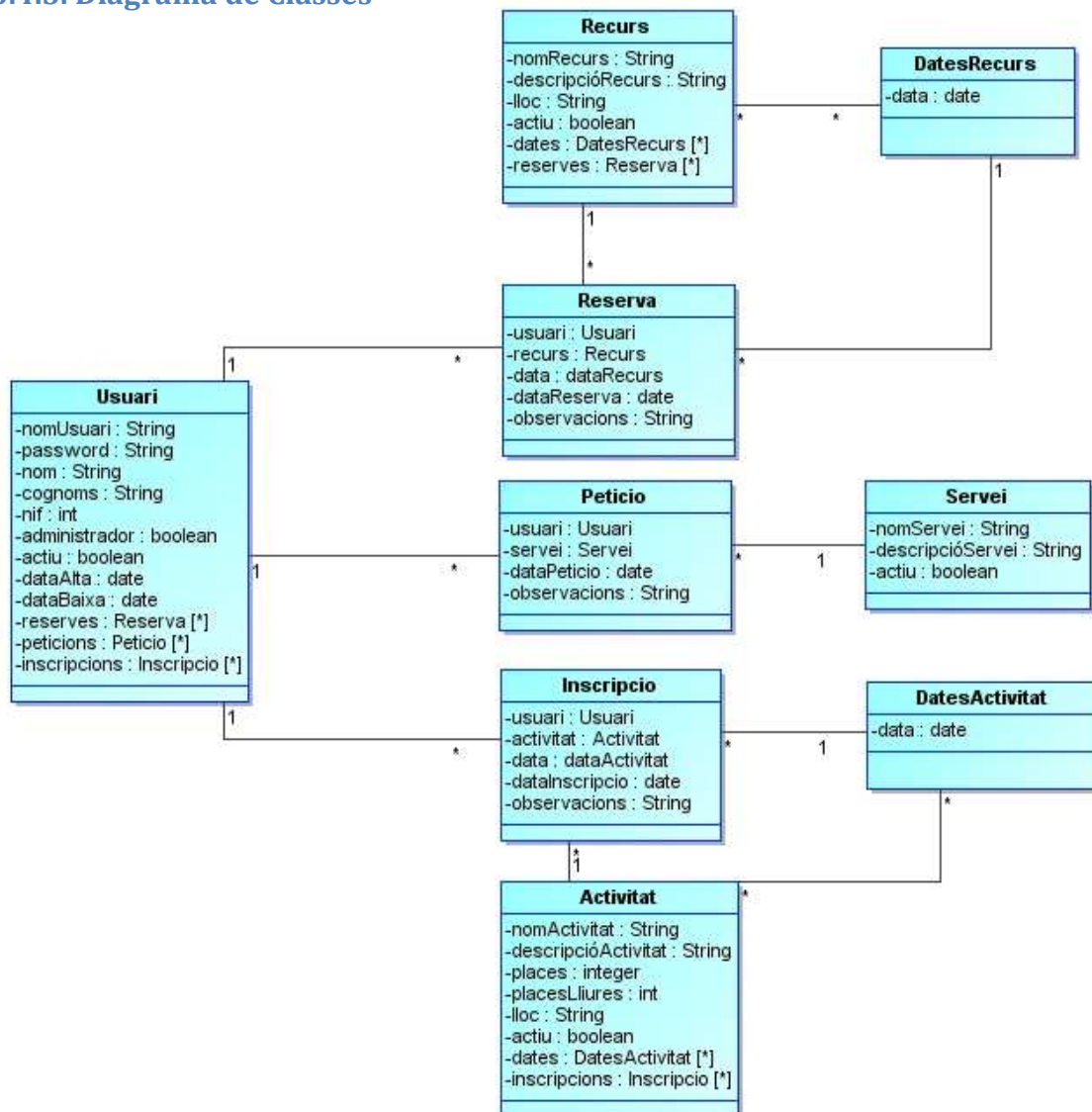


Figura 13. Diagrama de classes.

3.5. Altres diagrames

En aquest apartat podem veure com interactua l'usuari amb l'aplicació. Veurem l'enviament de missatges en alguns diagrames de seqüència.

3.5.1. Diagrama de seqüència Login

En aquest diagrama de seqüència veiem com l'usuari vol accedir a l'aplicació i com l'interfície d'usuari és comunica amb el servei WCF que accedeix a la base de dades per comprovar si les dades introduïdes són correctes:

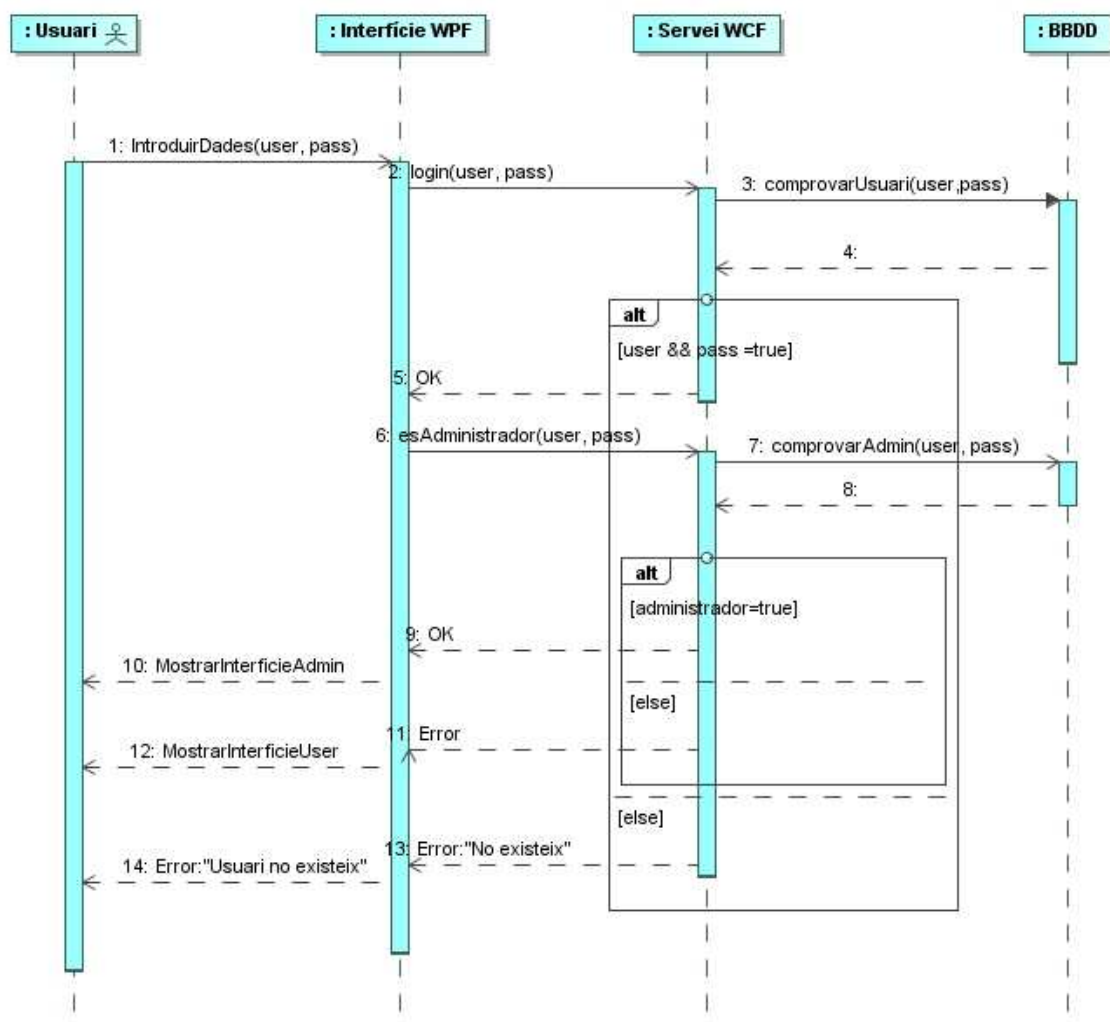


Figura 14. Diagrama de seqüència Login.

3.5.2. Diagrama de seqüència d'una Inscripció a una Activitat

En aquest diagrama veurem com un usuari consulta les activitats de l'hotel, i un cop mostrades per la interfície, l'usuari fa una inscripció. La reserva d'un recurs i la petició d'un servei serien molt semblants a la inscripció a una activitat.

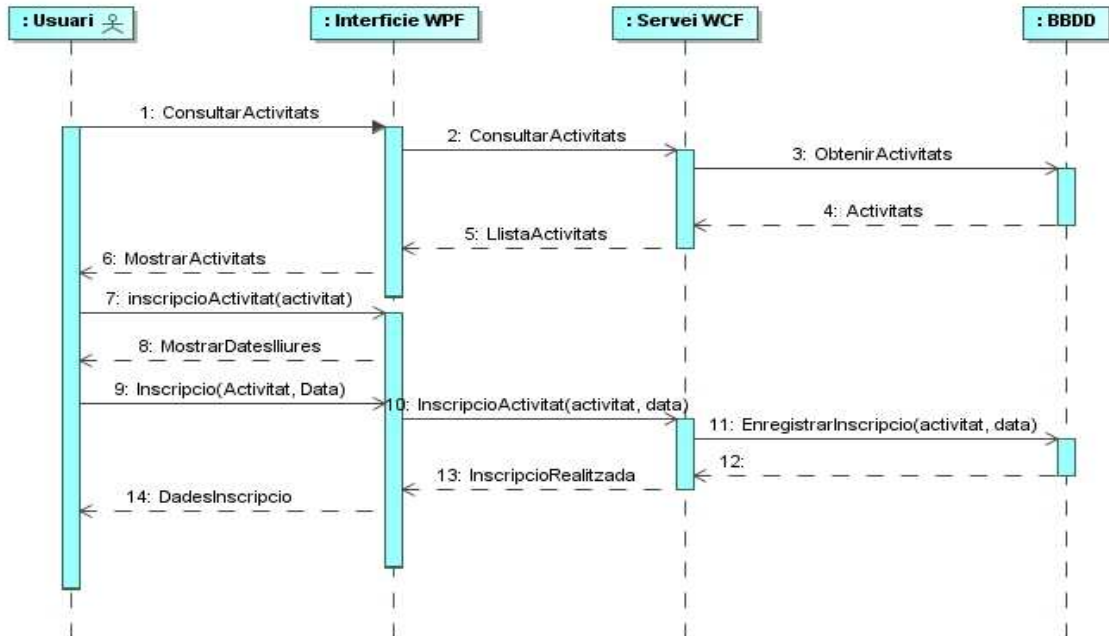


Figura 15. Diagrama de seqüència Inscripció Activitat.

3.5.3. Diagrama de seqüència Alta Servei

Al següent diagrama de seqüència veiem el que passa quan un administrador vol donar d'alta un servei. Seria molts semblant per donar d'alta una activitat, un recurs o un usuari.

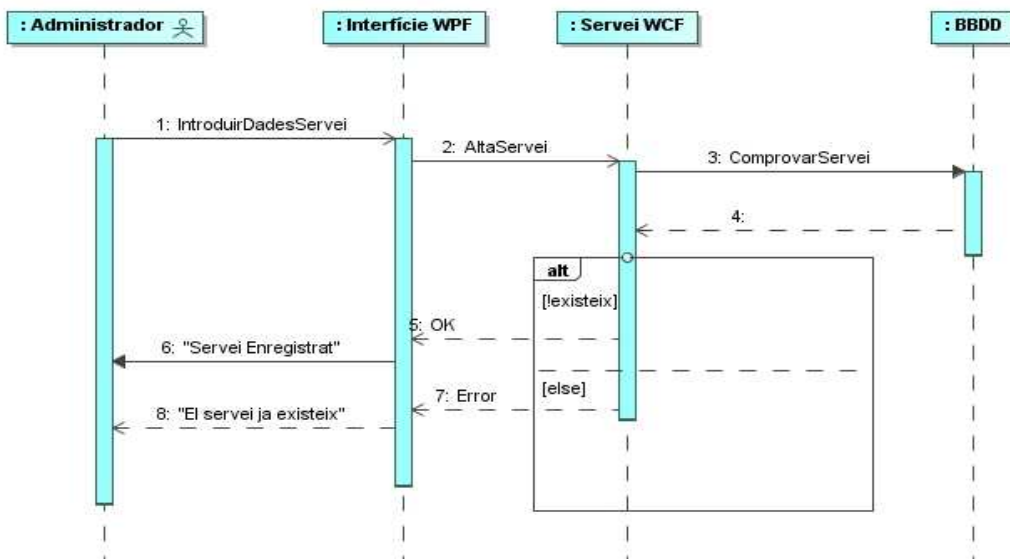


Figura 16. Diagrama de seqüència Alta Servei.

4. Disseny

4.1. Arquitectura de l'aplicació

A continuació descriurem l'arquitectura de la nostre aplicació. Primer veurem com és l'arquitectura física i , a continuació, l'arquitectura lògica.

4.1.1. Arquitectura física

L'arquitectura física emprada en el projecte ha estat del tipus client - servidor. L'hotel té terminals a totes les habitacions, a diferents punts d'interès pels clients i a la recepció i despatxos dels gestors. En aquests terminals instal·larem l'aplicació client que es connectarà amb el servidor. Al servidor es trobarà la base de dades i el servei WCF que permetrà treballar amb les dades. Podem veure un esquema a la següent figura:

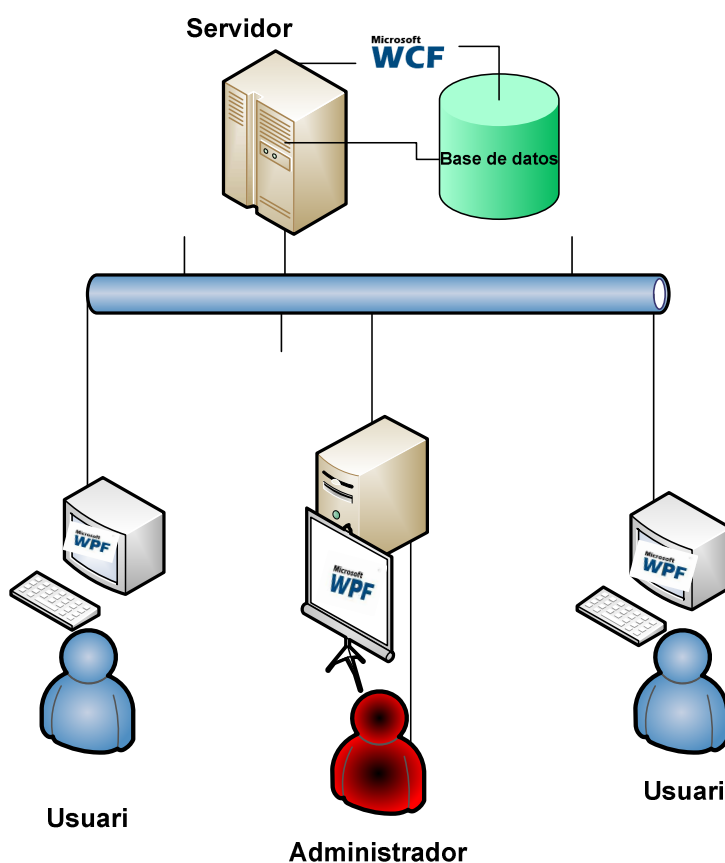


Figura 17. Arquitectura física.

4.1.2. Arquitectura lògica

Hem decidit dissenyar l'aplicació seguint el model de desenvolupament de tres capes:

- **Capa de presentació:** Aquesta capa mostra informació a l'usuari de l'aplicació i permet l'entrada de dades al sistema. La interfície d'usuari assumirà aquesta funcionalitat. La capa de presentació conté els components necessaris per la interacció entre l'usuari i l'aplicació. Es desenvoluparà amb *Windows Presentation Foundation*.
- **Capa de negoci:** Aquesta capa conté funcions de negoci. Hi trobarem els serveis necessaris encarregats de fer les connexions amb la base de dades, així com fer totes les comprovacions necessàries pel correcte funcionament de l'aplicació. Es pot dissenyar la lògica de les capes de negoci per ser utilitzades directament pels components de presentació o per ser encapsulada com un servei. En el nostre cas utilitzarem la segona opció, utilitzarem *Windows Communications Foundation* per crear els serveis que faran complir la lògica de negoci.
- **Capa de dades:** La capa de dades guarda les dades de manera persistent a la base de dades. Pràcticament totes les aplicacions necessiten enregistrar i accedir a algun tipus de dades. Utilitzarem un servidor *Microsoft SQL Server 2008* i *LINQ-to-SQL* per facilitar la relació entre les classes del programa i les taules de la base de dades.

4.2. Diagrama de desplegament

A continuació veiem com els diferents components de la nostre aplicació es comuniquen. Als diferents terminals dels clients trobarem l'aplicació d'escriptori WpfHotelSpa que es comunica amb el servidor on es troben la base de dades i el servei HotelSpaWCF. El servei HotelSpaWCF fa ús de la llibreria de classes LinqHotelSpa que és l'encarregada d'accedir directament a la base de dades HSPA que es troba al servidor Microsoft Sql Server 2008.

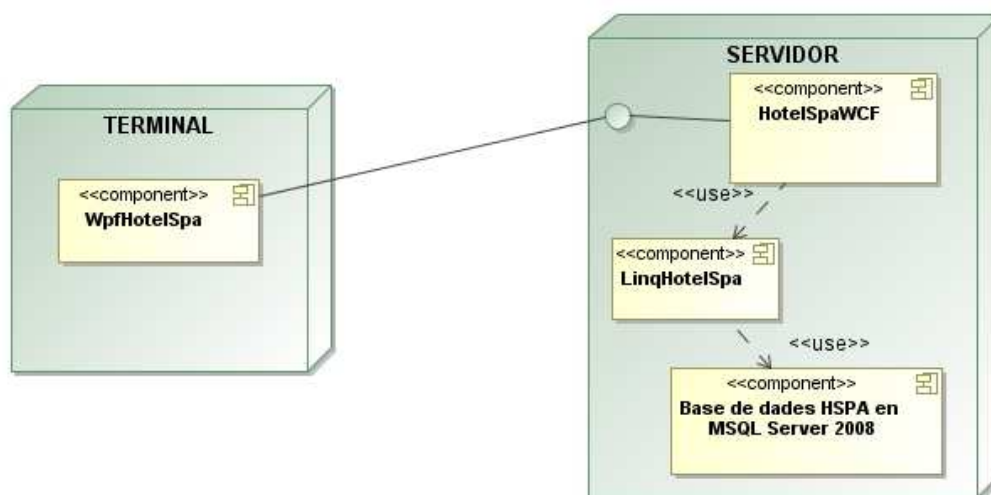


Figura 18. Diagrama de desplegament.

4.3. Disseny Base de Dades

Partint del diagrama de classes de partida obtenim el diagrama de la base de dades. S'ha creat un script amb Microsoft SQL Server Management Studio, anomenat **ScriptBaseDadesHSPA.sql**, per poder crear la base de dades i introduir algunes dades per provar l'aplicació.

4.3.1. Obtenció del disseny de les taules

- ❖ **Usuari**(NomUsuari, Password, Nom, Cognoms, Administrador, Actiu, DataAlta, DataBaixa);
- ❖ **Servei**(NomServei, DescripcioServei, Actiu);
- ❖ **Activitat**(NomActivitat, DescripcioActivitat, Lloc, Places, Actiu);
- ❖ **Recurs**(NomRecurs, DescripcioRecurs, Lloc, Actiu);
- ❖ **DatesActivitat**(NomActivitat, DataActivitat);
{NomActivitat} és clau forana a Activitat.
- ❖ **DatesRecurs**(NomRecurs, DataRecurs);
{NomRecurs} és clau forana a Recurs.

- ❖ **Peticio**(NomUsuari, NomServei, DataPeticio, DataRealitzacio, Realitzat, Observacions);
 {NomUsuari} és clau forana a Usuari.
 {NomServei} és clau forana a Servei.
- ❖ **Inscripcio**(NomUsuari, NomActivitat, DataActivitat, DataInscripcio, Observacions);
 {NomUsuari} és clau forana a Usuari.
 {NomActivitat} és clau forana a DatesActivitat.
 {DataActivitat} és clau forana a DatesActivitat.
- ❖ **Reserva**(NomUsuari, NomRecurs, DataRecurs, DataReserva, Observacions);
 {NomRecurs} és clau forana a DatesRecurs.
 {DataRecurs} és clau forana a DatesRecurs.

Ha diferència del plantejament inicial de la base de dades, he fet unes petites modificacions a les taules que s'especifiquen a continuació:

✓ **Taules Servei, Activitat i Recurs**

S'ha afegit l'atribut *Actiu* per indicar si un servei, una activitat o un recurs, que van ser donats d'alta en el seu moment, estan actius o no. Quan eliminem un servei, una activitat o un recurs, realment no els eliminem de la base de dades ja que perdríem tota la informació referent a peticions, inscripcions o recursos. D'aquesta forma, quan fem una eliminació, el que fem realment és posar a "False" l'atribut *Actiu* del registre eliminat. Si es vol tornar a donar d'alta una activitat l'aplicació ens permet recuperar-la canviant aquest valor a "True".

✓ **Taula Activitat**

Hem eliminat l'atribut *PlacesLliures* de la taula Activitat ja que aquest atribut és calculable. Quan un usuari vol realitzar una inscripció a una activitat en una data, el sistema consultarà el nombre d'inscripcions de l'activitat seleccionada en la data seleccionada i sabrà si queden places lliures o, en cas contrari, si no queden places.

✓ **Taula Reserva**

A la taula Reserves hem modificat la clau primària ja que, com estava plantejada en un principi, permetia que diferents usuaris reservessin un mateix recurs en una mateixa data. Amb la modificació, la clau primària estarà formada per *NomRecurs* i *DataRecurs* (claus foranes de la taula

DatesRecurs), de forma que si un usuari reserva un recurs en una data cap usuari més podrà fer la mateixa reserva.

4.3.2. Diagrama de la base de dades

A continuació veiem com queda el diagrama de la base de dades anterior un cop l'hem creada al Microsoft SQL Server 2008.

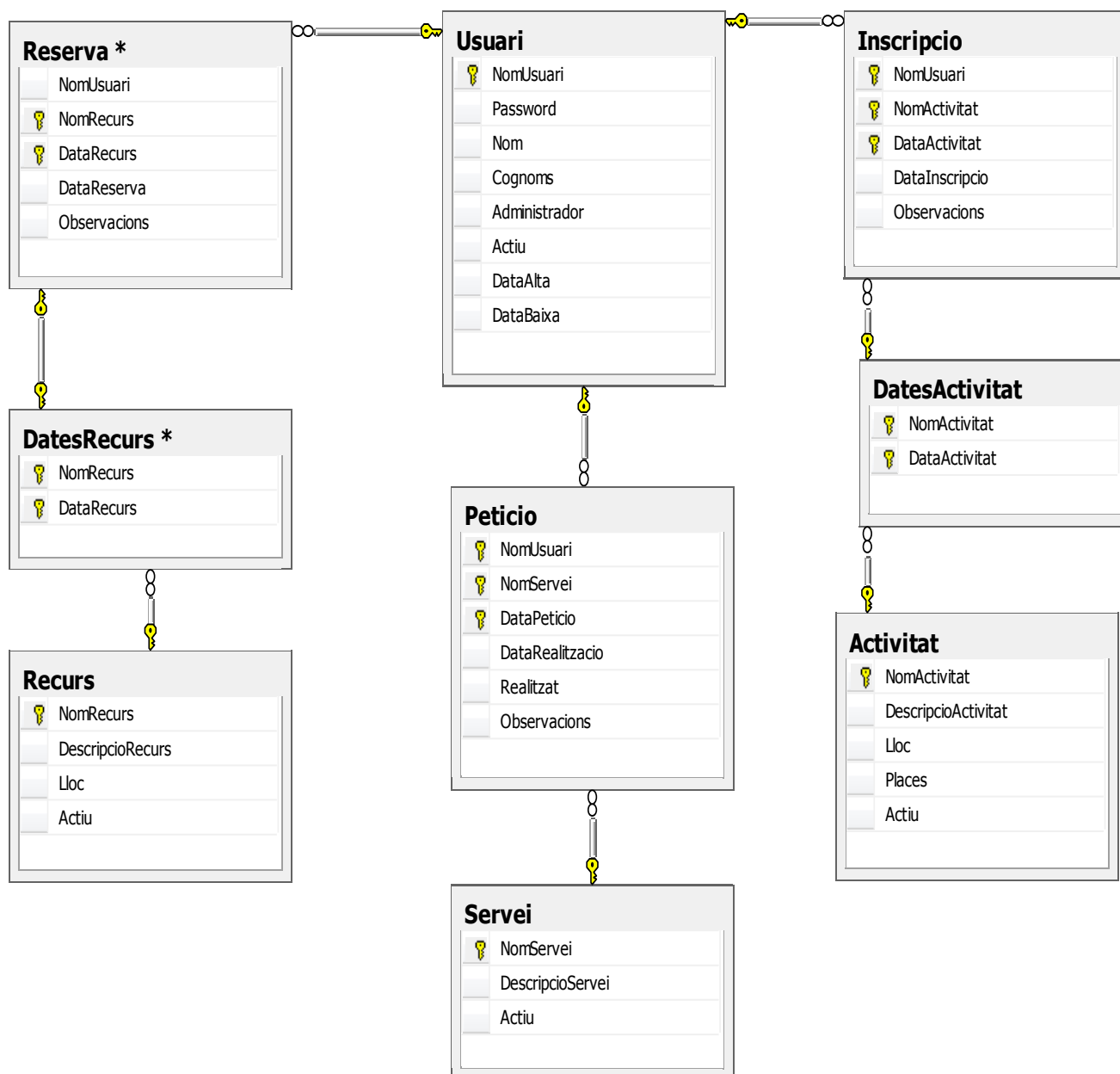


Figura 19. Diagrama base de dades.

4.4. Classes de LINQ-To-SQL

Partint del diagrama de base de dades anterior creat amb Microsoft SQL Server 2008 i, utilitzant LINQ-To-SQL des de Visual Studio 2008 SP1, obtenim el diagrama de classes d'entitat que utilitzarem a la nostre aplicació.

Per obtenir les classes amb LINQ tenim que afegir un nou element de dades anomenat 'Clases de LINQ to SQL', indicar on es troba la base de dades i automàticament ens generarà les classes necessàries.

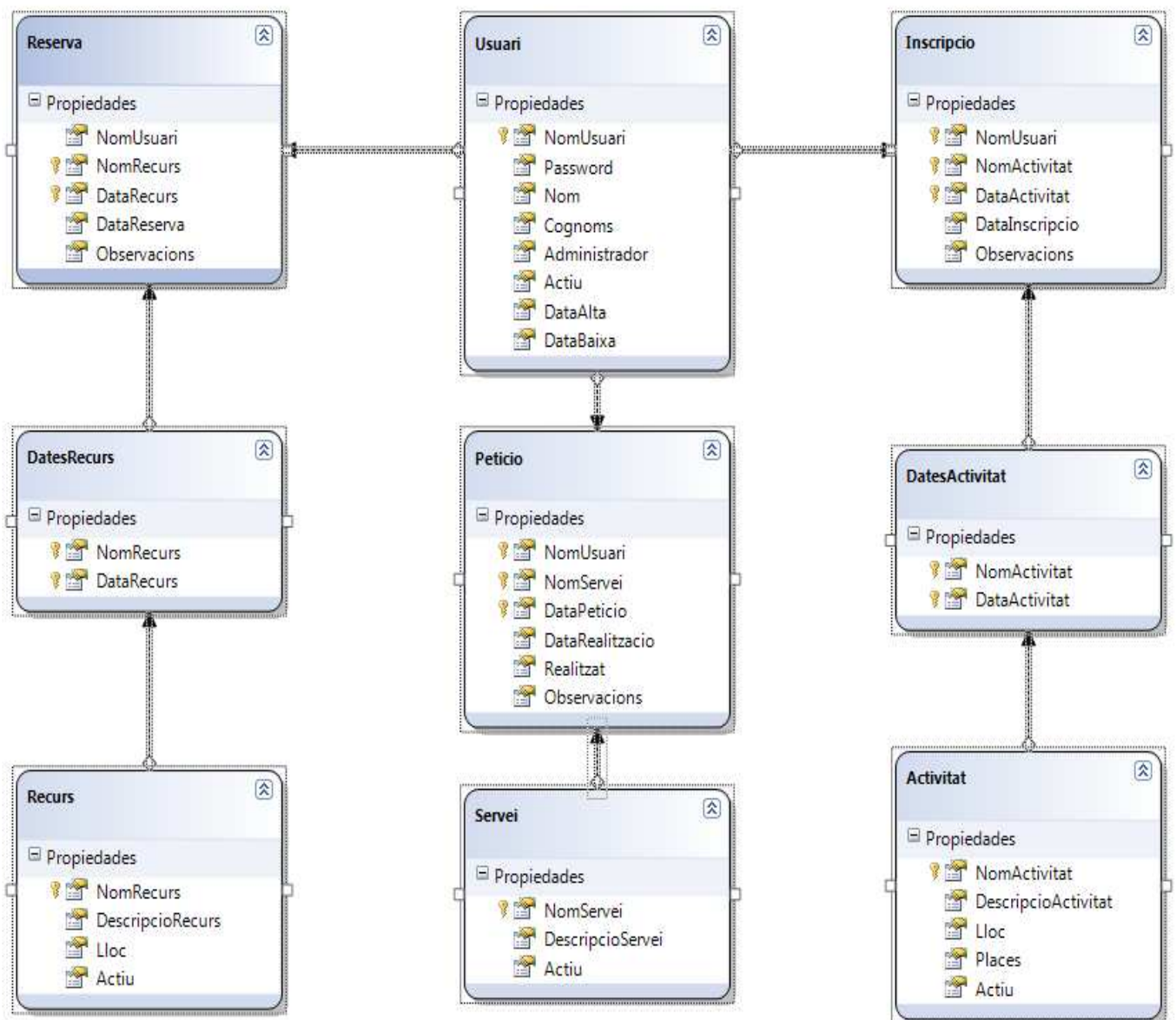


Figura 20. Diagrama de classe Linq To Sql.

5. Implementació

La fase de implementació del projecte ha estat la més dura ja que no havia tingut cap tipus de contacte amb la tecnologia .NET (C#, WPF, WCF, LINQ) ni amb Microsoft SQL Server 2008.

5.1 Etapes

Podríem dividir l'etapa d'implementació en 6 fases ben diferenciades:

1- Base de dades HSPA.mdf

Durant la fase d'anàlisi i disseny es va crear la base de dades i durant l'implementació es van modificar alguns atributs de les taules comentats a l'apartat [4.3.1. Obtenció del disseny de les taules](#).

2- Linq To Sql: Biblioteca de classes LinqHotelSpa

Aquesta part va ser la més senzilla. Primer es va crear un nou projecte de biblioteca de classes, a continuació es va afegir un nou element al projecte del tipus "Clases de LINQ to SQL". Per últim, només ens faltava afegir les taules de la nostre base de dades al dissenyador relacional d'objectes de Linq, per tal de que Linq To Sql implementés les classes per obtenir els objectes que s'utilitzarien en les següents fases per poder accedir a la base de dades, estalviant molta feina de programació.

Es pot veure com queda el diagrama de les classes d'entitat creat utilitzant LINQ To SQL a l'apartat [4.4. Classes de Linq To Sql](#).

3- Servei WCF: HotelSpaWCF.SpaService1

La creació del servei que permet a l'aplicació d'escriptori interactuar amb la base de dades és on van començar les dificultats. Després d'obtenir informació de diverses pàgines web vam entendre com funciona un servei WCF i es va començar la seva implementació.

Es va crear un nou projecte de "Biblioteca de Serveis WCF", ja que a l'aplicació client tindríem que afegir una referència al servei per tal de que l'aplicació el pogués consumir. Un cop realitzats els "DataContracts" corresponents a les operacions que el servei exposaria a les aplicacions que el volguessin consumir, es va afegir una referència a la llibreria de classes creada anteriorment, LinsHotelSpa.dll, per tal de que'l servei la utilitzés per poder treballar amb la base de dades.

4- Aplicació d'escriptori: WpfHotelSpa

Per crear l'aplicació d'escriptori i poder utilitzar els servei WCF és va afegir una referència de servei al projecte. També va ser necessari introduir una referència a la biblioteca de classes creada amb LINQ To SQL per a que l'aplicació pogués treballar amb els diferents objectes que li retornarien els mètodes del servei (objectes de tipus Usuari, Reserva, Peticio, Activitat,etc...)

Un cop preparat el projecte i dissenyades les finestres amb els controls necessaris, només faltava fer la implementació de l'aplicació y cridar als mètodes que ens ofería el servei per treballar amb les taules.

Va ser necessari instal·lar uns controls gratuïts, WPFToolKit de CodePlex, per poder utilitzar el seu DataGrid ja que .NET no disposa d'aquest control per utilitzar-lo amb WPF. Aquests controls es poden descarregar de <http://wpf.codeplex.com/releases/view/25047> .

Sempre que es crida un mètode del servei WCF des de l'aplicació WPF cridem un mètode per obrir la connexió amb el servei i un cop hem consumit els mètodes que ens interessin es crida un altre mètode per tancar-la.

5- Aplicació de consola: Servidor

Un cop es van fer les proves necessàries per comprovar que tot funcionava de la forma desitjada, ens vam donar compte de que faltava tenir el servei WCF escoltant peticions (i que no fos des del IDE del Visual Studio). Vam trobar diferents formes de fer-ho, com instal·lar el ISS(Internet Information Server) i configurar el servei al servidor. Finalment, vam optar per crear una senzilla aplicació de consola, Servidor, per hostatjar el servei. L'aplicació de consola el que fa és crear un HOST pel servei, obrir la consola i quedar-se a l'espera de que algú faci una petició al servei.

6- Instal·ladors de les aplicacions

Per últim, vam necessitar crear dos instal·ladors, un per l'aplicació de consola i un segon per l'aplicació d'escriptori. Aquesta part va ser molt senzilla ja que el Visual Studio 2008 va fer pràcticament tota la feina per nosaltres.

6. Treball futur

Algunes de les possibles modificacions de l'aplicació que podríem realitzar en un futur són les següents:

- **Especialitzar les funcionalitats de serveis, activitats i recursos.** Els serveis, activitats i recursos de l'hotel són tractats d'una forma bastant generalitzada i es podrien especialitzar més.
- **Automatització del servei de despertador.** El servei de despertador es podria automatitzar de forma que l'aplicació s'encarregués de trucar al client a la hora que especifiqués. Actualment no es possible ja que quan un client realitza una petició de despertador no té un camp per especificar l'hora, només per introduir observacions.
- **Eliminació persistent.** Introduir la funcionalitat de eliminar de forma persistent els usuaris, serveis, activitats o recursos. Actualment, quan volem eliminar un usuari, un servei, una activitat o un recurs, només podem deixar-los inactius per no perdre el seu historial. Es podria introduir una nova funcionalitat que donés la opció d'eliminar aquesta informació de manera persistent.

7. Conclusions

En aquest apartat comentaré l'experiència personal que m'ha suposat realitzar aquest Treball de Fi de Carrera.

Els objectius principals d'aquest Treball de Fi de Carrera eren, principalment, posar en pràctica tots els coneixements apresos a les diferents assignatures de l'Enginyeria Tècnica d'Informàtica de Sistemes i aprendre a utilitzar la tecnologia .NET.

En la meua opinió, els objectius principals han estat assolits, ja que ha sigut necessari posar en pràctica molts coneixements adquirits a diferents assignatures superades anteriorment com Base de dades I i II, Enginyeria del Programari, Programació Orientada a Objectes o Estructura de la Informació, entre d'altres.

Crear un projecte des del començament fins a la seva finalització, passant per totes les fases d'anàlisi i desenvolupament, ha estat una altre fita realitzada satisfactòriament.

Un cop finalitzat el projecte, els resultats obtinguts han estat molt positius, ja que quan vaig començar aquest TFC no coneixia cap de les tecnologies emprades i m'ha servit d'aprenentatge personal.

Finalment, després d'haver invertit moltes hores de treball, estudi i recerca, ha sigut gratificant poder concloure aquest treball i comprovar que es compleixen els requisits especificats en els objectius del projecte.

Bibliografia

Campderrich Falgueras, B.(2004) *Enginyeria del programari*. Barcelona: Fundació per a la Universitat Oberta de Catalunya.

Sistac Planas, J.(2005) *Base de dades I*. Barcelona: Fundació per a la Universitat Oberta de Catalunya.

Sistac Planas, J.(2004) *Base de dades II*. Barcelona: Fundació per a la Universitat Oberta de Catalunya.

Minguillón i Alfonso, J.(2003). *Programació orientada a l'objecte*. Barcelona: Fundació per a la Universitat Oberta de Catalunya.

Alvarez Canal, J.(2007). *Estructura de la informació*. Barcelona: Fundació per a la Universitat Oberta de Catalunya.

Moroney, L. (2008). *Silverlight 2.0*. Madrid: Anaya Multimedia.

Templeman, J; Vitter, D. (2002). *Visual Studio .NET* . Madrid: Anaya Multimedia.

MSDN Library (Español)

<[http://msdn.microsoft.com/es-es/library/ms123401\(v=MSDN.10\).aspx/](http://msdn.microsoft.com/es-es/library/ms123401(v=MSDN.10).aspx/)>

MSDN Academic Alliance

<http://msdn30.e-academy.com/uni_obertadc_inform/>

DevJoker.com. Linq To Sql – Un ejemplo sencillo

<<http://www.devjoker.com/contenidos/catss/326/LinQ-To-SQL--Un-ejemplo-sencillo.aspx/>>