

## **Memòria del projecte**

---

**Gestió acadèmica d'un centre de formació amb .NET**

**Jonathan Manjarrés Freire**

Estudis: Enginyeria Tècnica en Informàtica de Sistemes

Consultor: David Gañán Jiménez

11/01/2007

## **Dedicatòria i agraïments.**

Vull agrair en primer lloc al meu tutor David Gañán pel suport, recolzament i comprensió que he rebut per la seva part. Després d'un problema personal de gran indol creia que tot estava gairebé perdut, però amb els seus consells i comprensió m'ha sabut treure la motivació i compromís suficient per que aquest projecte pugui veure la llum. He de dir que no ha sortit tot como tenia planejat però crec que dades les circumstàncies donades durant el desenvolupament del projecte, puc treure el cap amb satisfacció per la tasca aconseguida.

En segon lloc, i no menys important, he de donar les gràcies a la meva família que m'han donat tot el suport i recolçament imaginables en aquest món per poder aconseguir finalitzar aquest TFC. Sobretot vull donar les gràcies a la meva dona, que es qui més ha patit les meves preocupacions i nervis en un moment difícil per a nosaltres. Es a ella a qui dedico principalment aquest projecte per les seves hores que compartim i me las ha cedit per aconseguir arribar a aquest moment.

Moltes gràcies als meus pacients amics i companys que m'han posat tot el més fàcil possible. Gràcies a totes les persones del meu voltant que d'una manera o d'altre han estat amb mi.

I sobretot no vull oblidar-me de tu, avi. Moltes gràcies per haver-me permès compartit part de la meva vida amb tu. Gràcies per haver donat la vida al meu pare per que ell me la va donar a mi, i es per això que dins meu tinc part de tu, cosa que em fa sentir orgullós. T'estimo.

Gràcies.  
Jonathan Manjarrés

## Resum

El projecte **Gestió acadèmica d'un centre de formació amb .NET** té com a finalitat informatitzar tot el procés de control de les matriculacions dels alumnes i gestió dels cursos que s'imparteixen i dels recursos del centre. Amb això es pretén aconseguir una solució completa que permeti fer el seguiment de totes les matriculacions als cursos dels alumnes, portar un manteniment exhaustiu dels cursos que s'imparteixen i fer un seguiment dels recursos del centre des de que es fa una petició fins que es fa servir.

El projecte consta de 2 aplicacions ben diferenciades on la primera conté 5 subaplicacions o mòduls y la segona conté 2:

- La primera aplicació es una aplicació Windows anomenada **GestColes** que permet al **secretari/a o personal administratiu** administrar 5 mòduls:

Un s'anomena **Gestió de Cursos** que permet gestionar cursos de que disposa el centre.

Un altre s'anomena **Gestió d'assignatures** que permet gestionar les assignatures que s'imparteixen al centre.

Altre s'anomena **Gestió Recursos** que permet gestionar els recursos de que disposa el centre.

Un altre s'anomena **Gestió Calendaris** que permet gestionar el calendari lectiu de cada any lectiu per al Centre.

Per últim tenim l'anomenat **Gestió d'usuaris** que permet gestionar els usuaris relacionats amb el centre com ara els alumnes i els professors.

- La segona aplicació es una aplicació WEB anomenada **GestColesWEB** que i consta de 2 mòduls i segons quin tipus d'usuari accedeixi a l'aplicació (Professor o Alumne) disposarà d'un mòdul o un altre:

Per un costat tenim el mòdul anomenat **Gestió Utilitats Alumne** on l'usuari alumne podrà gestionar peticions de tutories cap als professors i podrà visualitzar el seu expedient acadèmic. Només hi té accés l'usuari de tipus Alumne.

Per altre costat tenim el mòdul anomenat **Gestió Utilitats Professor** on l'usuari professor podrà gestionar peticions de recursos acadèmics, introduir qualificacions dels alumnes i gestionar les peticions de tutoria que li arriben. Només hi té accés l'usuari de tipus Professor.

L'aplicació s'ha desenvolupat completament utilitzant la plataforma .NET 1.0 amb el programa de desenvolupament Visual Studio 2002 Service Pack 1. El llenguatge de programació utilitzat ha estat C# .NET (versió 2002) i s'ha fet servir les tecnologies d'Aplicació Windows, Aplicació WEB amb ASP.NET i accés a dades mitjançant ADO.NET.

# Índex de continguts

<b>1. Introducció</b>	<b>8</b>
1.1 Justificació del TFC i context en el qual es desenvolupa: punt de partida i aportació del TFC	8
1.2 Objectius del TFC	10
1.3 Enfocament i mètode seguit	11
1.4 Planificació del projecte	12
1.5 Productes obtinguts	14
1.6 Descripció dels altres capítols de la memòria	15
<b>2. Arquitectura del projecte</b>	<b>16</b>
2.1 Introducció	16
2.2 Descripció de les funcionalitats	16
2.2.1 Components del projecte	16
2.2.2 Descripció del funcionament	18
2.3 Requeriments funcionals	19
2.4 Escenari del projecte	23
2.5 Arquitectura de les aplicacions del projecte	24
2.5.1 Aplicació Windows	24
2.5.2 Aplicació WEB	25
2.6 Disseny de la base de dades	26
2.6.1 Disseny conceptual	26
2.6.2 Disseny lògic	27
2.6.3 Base de dades GestColes	29
2.7 Disseny d'estructura de capes de les aplicacions	29
2.8 Disseny de classes de les aplicacions	31
2.9 Seguretat i accés dels usuaris	37
2.9.1 Aplicació Windows	37
2.9.2 Aplicació WEB	37
2.10 Requisits tècnics de maquinari i programari	38
<b>3. Descripció de la solució</b>	<b>39</b>
3.1 Mòdul Windows	39
3.1.1 Introducció	39
3.1.2 Inici	39
3.1.3 Gestió Cursos	40
3.1.3.1 Assignar Calendaris	40
3.1.3.2 Assignar Assignatures	41
3.1.3.2.1 Assignar Professor	41
3.1.3.2.2 Assignar Aula	42
3.1.4 Gestió Assignatures	42
3.1.5 Gestió Calendaris	43
3.1.6 Gestió Recursos	43
3.1.6.1 Petició de recurs	44
3.1.7 Gestió Usuaris	44
3.1.7.1 Matricular Alumne	45
3.1.8 Finestres	45
3.1.9 Sortir	45
3.2 Mòdul WEB	46
3.2.1 Introducció	46
3.2.2 Inici	46
3.2.3 Inici Sessió	46

3.2.4	Gestió Utilitats Alumne	47
3.2.4.1	Petició Tutoria Alumne	47
3.2.4.1.1	Visualitzar Expedient	48
3.2.5	Gestió Utilitats Professor	49
3.2.5.1	Petició Recurs	49
3.2.5.2	Petició Tutoria	50
3.2.5.3	Qualificació Alumne	50
<b>4.</b>	<b>Línies de desenvolupament futur</b>	<b>51</b>
<b>5.</b>	<b>Valoració econòmica del projecte</b>	<b>52</b>
<b>6.</b>	<b>Conclusions</b>	<b>53</b>
<b>7.</b>	<b>Glossari</b>	<b>54</b>
<b>8.</b>	<b>Bibliografia i referències</b>	<b>55</b>
<b>9.</b>	<b>Annexos.</b>	<b>56</b>
9.1	Consideracions de les aplicacions del projecte	56

# Índex de figures

Figura 1: Taula amb els diferent mòduls de l'aplicació Windows del projecte	9
Figura 2: Taula amb els diferent mòduls de l'aplicació ASP.NET (WEB) del projecte	9
Figura 3: Diagrama detallat del marc de treball .NET (.NET Framework)	9
Figura 4: Taula amb els serveis i tecnologia emprats en el projecte	10
Figura 5: Fases del desenvolupament del projecte	11
Figura 6: Taula amb els documents i les seves dates de lliurament.	12
Figura 7: Diagrama de Gantt del projecte	13
Figura 8: Taula amb els productes obtinguts durant el desenvolupament del projecte	14
Figura 9: Mòduls de les aplicacions que formen el projecte	17
Figura 10: Cas d'Ús Mòdul Gestió Cursos	19
Figura 11: Cas d'Ús Mòdul Gestió Assignatures	20
Figura 12: Cas d'Ús Mòdul Gestió Recursos	20
Figura 13: Cas d'Ús Mòdul Gestió Calendaris	21
Figura 14: Cas d'Ús Mòdul Gestió d'Usuaris	21
Figura 15: Cas d'Ús Mòdul Gestió Utilitats Alumne	22
Figura 16: Cas d'Ús Mòdul Gestió Utilitats Professor	22
Figura 17: Escenari d'utilització del projecte	23
Figura 18: Arquitectura Client - Servidor de l'aplicació Windows del projecte	24
Figura 19: Arquitectura Client - Servidor de l'aplicació WEB del projecte	25
Figura 20: Model ER de la base de dades del projecte	26
Figura 21: Transformació del model ER en relacions	27
Figura 22: Diagrama lògic de la base de dades del projecte	28
Figura 23: Esquema de l'estructura de capes del projecte.	30
Figura 24: Mètodes de la classe <i>Curs</i> de la biblioteca de classes <i>AccesoDatos</i>	32
Figura 25: Mètodes de la classe <i>Assignatura</i> de la biblioteca de classes <i>AccesoDatos</i> .	33
Figura 26: Mètodes de la classe <i>Calendari</i> de la biblioteca de classes <i>AccesoDatos</i> .	33
Figura 27: Mètodes de la classe <i>Recurs</i> de la biblioteca de classes <i>AccesoDatos</i>	34
Figura 28: Mètodes de la classe <i>Usuari</i> de la biblioteca de classes <i>AccesoDatos</i>	35
Figura 29: Mètodes de la classe <i>Aula</i> de la biblioteca de classes <i>AccesoDatos</i>	35
Figura 30: Mètodes de la classe <i>Combo</i> de la biblioteca de classes <i>AccesoDatos</i>	35
Figura 31: Mètodes de la classe <i>Comun</i> de la biblioteca de classes <i>AccesoDatos</i>	35
Figura 32: Mètodes de la classe <i>Funciones</i> del projecte Windows <i>GestColes</i>	36
Figura 33: Mètodes de la classe <i>Funciones</i> del projecte ASP.NET (WEB) <i>GestColesWEB</i>	36
Figura 34: Menú Principal de l'aplicació Windows	39
Figura 35: Mòdul Gestió Cursos de l'aplicació Windows	40

Figura 36: Opció Assignar Calendaris de l'aplicació Windows	40
Figura 37: Opció Assignar Assignatures de l'aplicació Windows	41
Figura 38: Opció Assignar Professor a unes Assignatures de l'aplicació Windows	41
Figura 39: Opció Assignar Aula a unes Assignatures de l'aplicació Windows	42
Figura 40: Opció Gestió Assignatures de l'aplicació Windows	42
Figura 41: Opció Gestió Calendaris de l'aplicació Windows	43
Figura 42: Opció Gestió Recursos de l'aplicació Windows	43
Figura 43: Opció Assignació Assignatures a un professor de l'aplicació Windows	44
Figura 44: Opció Gestió Usuaris de l'aplicació Windows	44
Figura 45: Opció Matriculació Alumne de l'aplicació Windows	45
Figura 46: Opció Finestres de menú de l'aplicació Windows	45
Figura 47: Finestra de Login d'usuari de l'aplicació ASP.NET (WEB)	46
Figura 48: Finestra de Gestió d'utilitats d'alumne de l'aplicació ASP.NET (WEB)	47
Figura 49: Finestra de l'opció de Visualitzar Expedient de l'aplicació ASP.NET (WEB)	48
Figura 50: Finestra de Gestió d'utilitats de Professor de l'aplicació ASP.NET (WEB)	49
Figura 51: Finestra de l'opció de Gestió Peticions de Tutoria de l'aplicació ASP.NET (WEB)	50
Figura 52: Finestra de l'opció de Qualificar Alumne de l'aplicació ASP.NET (WEB)	50

# 1.Introducció

## 1.1 Justificació del TFC i context en el qual es desenvolupa: punt de partida i aportació del TFC

---

L'objectiu del projecte es la realització d'una solució completa per gestionar informàticament un centre de formació i permetre als usuaris de tipus Alumne i Professor disposar d'unes eines per consultar dades rellevants al seu desenvolupament. Això comporta una actualització a nivell d'impresos i dades que passen del paper tradicional a sistemes digitals d'emmagatzematge encara que a l'hora es faci servir per algunes qüestions una còpia en paper. Una solució completa com aquesta comporta resoldre una sèrie de problemes. Els mes importants son:

Informatitzar el departament de secretaria amb una aplicació molt intuïtiva i que no requereixi una formació especial per fer-la servir.

Promoure a l'alumnat i als professors de fer servir les noves Tecnologies, especialment Internet, per poder gestionar les eines i consultar tota la informació que el centre posi al seu abast des de qualsevol lloc i en qualsevol moment.

El projecte presentat intenta resoldre aquests problemes proporcionant una solució composta per 2 aplicacions on cadascuna d'elles va destinada a un entorn de treball específic (com als PC's de Secretaria o per Internet des de qualsevol PC) creades amb diferent tecnologia (Formularis Windows i ASP.NET) depenent als usuaris als que estan orientats.

Les dues aplicacions estan dividides a la vegada en varis mòduls encarregats de proporcionar una sèrie d'utilitats o eines a l'usuari. Les característiques principals d'aquests mòduls les podem veure reflectides a les taules següents:

La primera aplicació basada en Formularis Windows està dividida en 5 mòduls, alguns dels quals contenen diferents opcions per l'usuari, i requereix accés a la base de dades del projecte.

Mòdul	Usuaris	Descripció
Gestió Cursos	Secretari o personal administratiu	Permet a l'usuari portar el manteniment dels cursos que s'imparteixen al centre així com les relacions amb els calendaris lectius on s'hi oferta, les assignatures que formen part de cada curs, l'aula on s'imparteix cadascuna i el professor que imparteix cadascuna.
Gestió Assignatures	Secretari o personal administratiu	Permet a l'usuari portar el manteniment de les assignatures que s'imparteixen al centre.
Gestió Calendaris	Secretari o personal administratiu	Permet a l'usuari portar el manteniment dels calendaris lectius de cada any indicant les dades d'inici i fi i de les dates on el centre romandrà tancat.
Gestió Recursos	Secretari o personal administratiu	Permet a l'usuari portar el manteniment dels recursos de que disposa el centre de formació. A més permet a l'usuari gestionar les peticions que fan els professors sobre aquests recursos.



Gestió Usuaris	Secretari o personal administratiu	Permet a l'usuari portar el manteniment dels usuaris (de tipus professor i alumne) que passen pel centre de formació. A més permet el manteniment de les matricules dels alumnes.
----------------	------------------------------------	--

Figura 1: Taula amb els diferent mòduls de l'aplicació Windows del projecte

La segona aplicació basada en ASP.NET (WEB) està dividida en 2 mòduls, alguns dels quals contenen diferents opcions per l'usuari.

Mòdul	Usuaris	Descripció
Gestió Utilitats Alumne	Alumnes	Permet a l'usuari realitzar el manteniment de les seves peticions de tutoria i també visualitzar el seu expedient acadèmic actualitzat.
Gestió Utilitats Professor	Professors	Permet a l'usuari realitzar el manteniment de les seves peticions de recursos, la gestió de les peticions de tutoria que li fan els seus alumnes i qualificar als seus alumnes.

Figura 2: Taula amb els diferent mòduls de l'aplicació ASP.NET (WEB) del projecte

Per a la realització del projecte s'utilitzarà la plataforma .NET versió 1.0 que permet desenvolupar totes dues aplicacions amb els seus corresponents mòduls des d'un mateix entorn de treball (IDE) que en aquest cas serà Microsoft Visual Studio 2002 Service Pack 1.

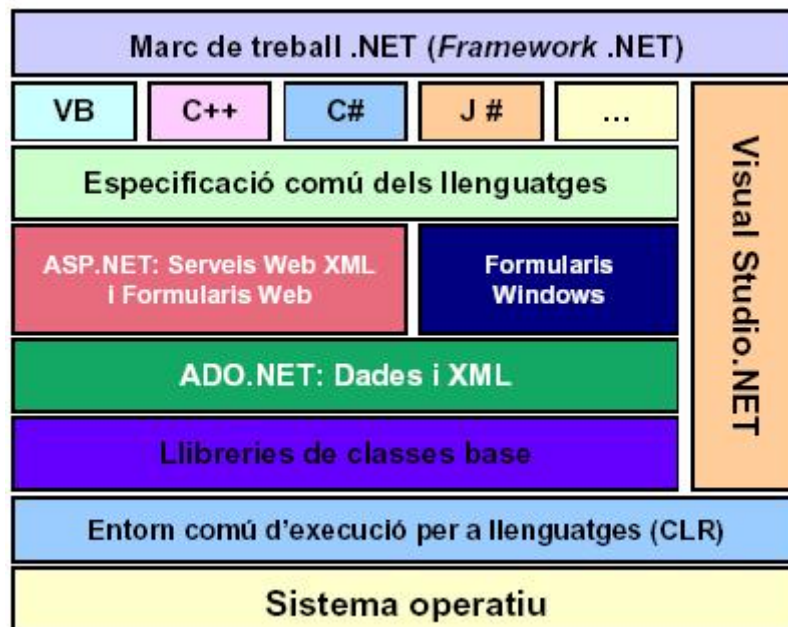


Figura 3: Diagrama detallat del marc de treball .NET (.NET Framework)

La plataforma .NET es basa en el marc de treball .NET (.NET Framework) creat per Microsoft. Es una capa de software intermedi que es situa entre les aplicacions que desenvolupem i el sistema operatiu. La figura 3 mostra l'estructura d'aquest marc de treball .NET. Es poden utilitzar diferents llenguatges de programació dins aquest marc de treball. Per aquest projecte utilitzaré el llenguatge C# .NET 2002.

La elecció d'aquest llenguatge es perquè ja tinc coneixement d'aquest en aquesta mateixa versió i perquè de tots els llenguatges oferts pel Visual Studio 2002 es el més potent i flexible.

El projecte utilitzarà un ampli ventall dels serveis proporcionats per la plataforma .NET que apareixen a les figures 3. La taula de la Figura 4 mostra els serveis i tecnologies utilitzades en cadascun dels mòduls de les dues aplicacions que consta el projecte.

<b>Aplicació</b>	<b>Serveis i Tecnologia</b>
Gestió Cursos	Framework.NET, ADO.NET, Formularis Windows
Gestió Assignatures	Framework.NET, ADO.NET, Formularis Windows
Gestió Calendaris	Framework.NET, ADO.NET, Formularis Windows
Gestió Recursos	Framework.NET, ADO.NET, Formularis Windows
Gestió Usuaris	Framework.NET, ADO.NET, Formularis Windows
Gestió Utilitats Alumne	Framework.NET, ASP.NET, ADO.NET, Formularis WEB
Gestió Utilitats Professor	Framework.NET, ASP.NET, ADO.NET, Formularis WEB

**Figura 4: Taula amb els serveis i tecnologia emprats en el projecte**

L'aportació d'aquest TFC es presentar un projecte on es pot trobar per una part una solució comercial completament funcional i operativa preparada per poder-la implantar en un centre de formació i, per altra banda, un producte on s'integren un ventall de les tecnologies presents al marc de treball .NET.

El projecte, un cop finalitzat, permetrà tenir una solució completa per a gestionar un centre de formació. A més, mitjançant uns petits canvis de imatge, es pot personalitzar adaptant-lo a un altre centre de formació.

Per a fer la presentació del producte més atractiva i propera a una solució comercial personalitzada, s'ha dissenyat per a un centre de formació imaginari mantenint un estil i disseny coherent en tots els mòduls de les dues aplicacions del projecte.

## **1.2 Objectius del TFC**

---

Es pot considerar com l'objectiu principal d'aquest projecte TFC el desenvolupament d'una aplicació completa i funcional amb un caire professional fent servir la plataforma .NET i emprant la major part de les seves tecnologies més importants (ASP.NET, ADO.NET, formularis WEB i Windows).

D'altra banda, la consecució d'aquest objectiu primordial comporta l'assoliment d'altres fites també molt importants com son:

- Aprendre a realitzar una aplicació seguint totes les etapes del procés de desenvolupament: anàlisi, disseny, implementació, depuració i prova.
- Conèixer l'arquitectura .NET.
- Conèixer les principals classes que apareixen al .NET Framework.
- Conèixer en profunditat el llenguatge C# .NET 2002.
- Adquisició de coneixements i experiència en un entorn de desenvolupament nou i cada cop més present en el mercat laboral.
- Aprendre a utilitzar l'entorn de treball Visual Studio 2002 Service Pack 1.
- Desenvolupament de components nous reutilitzables per altres aplicacions.
- Coneixement de Microsoft SQL Server 2000 com a sistema de gestió de bases de dades.
- Aprendre a crear programes d'instal·lació professionals per a distribuir i instal·lar les aplicacions desenvolupades.

### 1.3 Enfocament i mètode seguit

---

L'enfocament i el mètode seguit per a la realització d'aquest projecte es correspon amb el cicle de vida clàssic d'un projecte (també anomenat en cascada) i amb prototip. Si bé, donat a que l'aplicació és un TFC, hi haurà etapes com per exemple el manteniment que no es portaran a terme.

Així, les fases en que es dividirà el projecte es poden veure a la Figura 5 on s'inclouen els documents que seran les bases de partida per a les etapes següents. Realment no aplicarem un cicle de vida estrictament en cascada doncs aquesta filosofia de treball no permet tornar una fase enrere un cop s'ha completat mentre que aquest projecte ha estat obert a rectificacions, complements, ampliacions i canvis durant tota la seva realització.

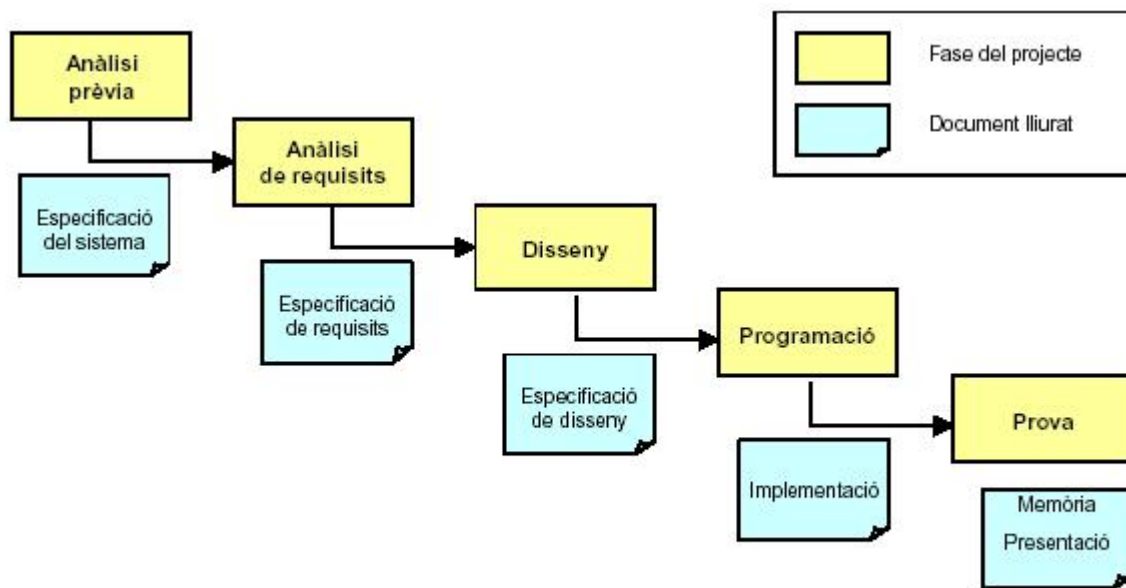


Figura 5: Fases del desenvolupament del projecte.

## 1.4 Planificació del projecte

---

La planificació seguida al projecte ve marcada per les dates de lliurament definides al calendari de l'assignatura. A la figura 6 es mostra una taula amb les dates clau així com els documents que es lliuraran en cadascuna d'aquestes dates.

Data Lliurament	Document	Descripció del document
29-setembre-06	<b>PAC 1</b> Anàlisi prèvia	És el pla de treball on es presenta una descripció general i la planificació temporal del projecte.
15-octubre-06	Anàlisi de requisits	Document que especifica amb suficient precisió que farà el programari creat per posteriorment desenvolupar-lo.
31-octubre-06	<b>PAC 2</b> Disseny	Determina l'estructura de la base de dades, les classes i les interfícies que tindrà l'aplicació.
18-decembre-06	<b>PAC 3</b> Implementació	L'aplicació final completa i funcional.
11-gener-07	Memòria i presentació virtual	A la memòria del projecte es sintetitzarà el treball realitzat en el TFC i es mostrarà l'assoliment dels objectius proposats.  La presentació virtual resum tot el contingut de la memòria virtual.

Figura 6: Taula amb els documents i les seves dates de lliurament.

A la figura 7 es pot veure el diagrama de Gantt corresponent a la planificació del projecte. Es mostren les fases del projecte amb el desglossament en les seves respectives subtasques. Per a cada tasca i subtasca s'especifica la data d'inici i finalització així com la seva duració estimada en dies.

Es pot observar al diagrama de Gantt com les fases del projecte s'han planificat de manera que no es solapin seguint el model en cascada.

Només en el cas de les tasques corresponents a l'elaboració de la memòria final i de la presentació virtual s'ha considerat que es realitzaran a la vegada per aprofitar el temps d'una manera òptima.

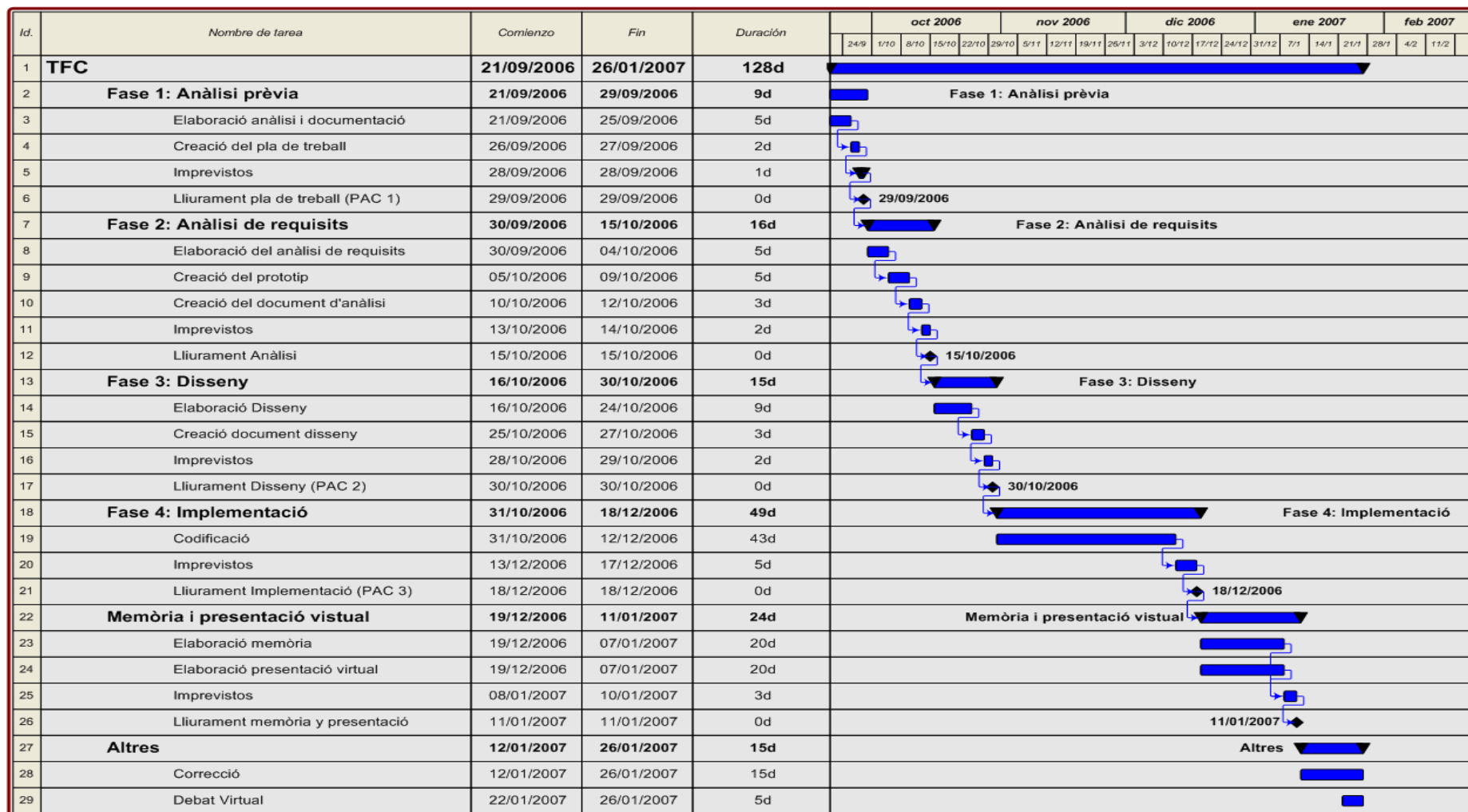


Figura 7: Diagrama de Gannt del projecte.

## 1.5 Productes obtinguts

---

Els productes obtinguts són els típics que s'obtenen si es segueix el model clàssic d'elaboració d'un projecte amb l'afegit d'un prototip. La Figura 8 mostra una relació de tots aquests productes obtinguts en el desenvolupament del projecte.

Producte	Descripció
<b>Anàlisi prèvia (Pla de treball)</b>	Descripció inicial del projecte on s'indiquen els objectius i es presenten els requeriments funcionals i tècnics necessaris.
<b>Anàlisi de requisits</b>	El document d'anàlisi inclou una descripció detallada dels components del projecte i del seu funcionament.  Es mostren els requeriments funcionals de cadascuna de les aplicacions que componen el projecte mitjançant la descripció dels casos d'ús existents.  També es mostren els diagrames de seqüència per a cada cas d'ús.
<b>Disseny</b>	El document de disseny inclou una descripció del projecte que apunta als components del projecte i a l'escenari on s'utilitzarà cadascuna de les aplicacions.  També conté una visió general de l'arquitectura de les aplicacions del projecte, els diagrames de flux de les interfícies d'usuari, el disseny de l'estructura per capes d'aquestes aplicacions, el disseny de les classes principals del projecte i el disseny de la base de dades que s'utilitza.
<b>Implementació</b>	Es lliura les aplicacions del projecte amb cada mòdul que les componen de manera que es pugui testejar d'una manera més real la interfície de cada aplicació. Es lliura en un fitxer ZIP on anirà el codi font, executables, imatges, Base de Dades i tots els fitxers que formin part de la solució del projecte.
<b>Memòria del projecte</b>	És aquest document.
<b>Presentació del projecte</b>	Presentació multimèdia complementària a la memòria del projecte que fa èmfasi en els punts més importants donant una visió global del projecte.
<b>Codi font</b>	Si cal, es lliuren tots els arxius de la solució Visual Studio 2002 del projecte comprimits en un fitxer ZIP amb les modificacions i/o adicions d'última hora, a més de la base de dades utilitzada per les proves.
<b>Producte final</b>	Optativament es pot crear un instal·lador de les aplicacions del projecte i lliurar-lo.

Figura 8: Taula amb els productes obtinguts durant el desenvolupament del projecte.

## **1.6 Descripció dels altres capítols de la memòria**

---

A continuació es fa un petit resum del contingut de la resta de capítols de la memòria.

**Capítol 2: Arquitectura del projecte.** Aquest capítol presenta una descripció detallada dels aspectes funcionals i estructurals del projecte. Es fa una descripció de l'escenari on es situa el projecte i de les arquitectures involucrades en cadascun dels mòduls de les dues aplicacions desenvolupades. A continuació es mostra un disseny de la base de dades del projecte, l'estructura de capes en que es munta la aplicació i de les classes i mètodes més importants que apareixen.

Tot seguit es parla dels components utilitzats en el projecte així com d'un component desenvolupat expressament per a aquest projecte. Per a finalitzar es tracta el tema de seguretat i accés dels usuaris a les dues aplicacions del projecte i dels requisits a nivell hardware i software necessaris per poder instal·lar i posar en funcionament el projecte en un escenari real.

**Capítol 3: Descripció de la solució.** Aquest capítol presenta una descripció de les aplicacions que componen el projecte i de les seves funcionalitats.

Els últims capítols presenten una valoració econòmica del projecte incloent els costos de desenvolupament i d'implantació en un escenari real. Una conclusió final, les possibles línies de desenvolupament futur per millorar les aplicacions, un glossari de termes utilitzats i la bibliografia i referències d'Internet que s'han fet servir per al desenvolupament d'aquest TFC.

## 2.Arquitectura del projecte

### 2.1 Introducció

---

El projecte gestió d'un centre de formació amb .NET té com a finalitat informatitzar tots els processos per posar en marxa un curs i els processos per a la matriculació dels alumnes a aquests cursos. Amb això es pretén aconseguir una solució completa que permeti controlar des de la creació d'un nou calendari lectiu fins a que un alumne es matricula a un curs i per altre costat permetre el control de diverses opcions i informacions que els usuaris de tipus professor o alumne podran gestionar des d'Internet.

A continuació es mostren en profunditat els aspectes més rellevants d'aquest projecte. Primerament es veu una descripció de les dues aplicacions que conté i, seguidament de cadascuna, una descripció dels diferents mòduls de que es componen, quin tipus d'usuaris els utilitzaran i com i quan s'utilitzen aquests durant el treball que es realitza a un centre de formació.

### 2.2 Descripció de les funcionalitats

---

#### 2.2.1 Components del projecte

El projecte conté dues aplicacions ben diferenciades. Una aplicació Windows i una aplicació WEB. A continuació desglossarem els components de ambdós.

L'aplicació Windows anomenada **GestColes** consta de 5 mòduls ben diferenciats. Cadascun d'ells està orientat a realitzar una determinada tasca i estan pensats per a que l'utilitzi l'usuari Secretari/a o personal administratiu. Com que les gestions que poden realitzar estan ben diferenciades, s'han dissenyat els diferents mòduls de l'aplicació que faran servir els usuaris pensant en les necessitats i funcionalitats de cada tasca. L'usuari Secretari/a i/o personal administratiu requereixen una aplicació visual fàcil d'entendre i que automatitzi tots els processos possibles.

Un mòdul es el de **Gestió de cursos** que permet a l'usuari crear, modificar o eliminar un curs al sistema i a més permet compondre els cursos *actius* amb el calendari lectiu on s'impartiran, les assignatures de que disposa el centre que constaran al pla docent dels cursos. També permet per a cada assignatura que s'imparteix a un curs indicar a quina aula s'impartirà i que professor l'imparteix.

Altre mòdul es el de **Gestió de calendaris** que permet a l'usuari crear, modificar o eliminar un calendari lectiu al sistema. Els calendaris lectius a més contenir la data d'inici i fi també contenen informació sobre dates on el centre romandrà tancat com per exemple festius Nacionals i Locals. Aquests calendaris son essencials per poder compondre un curs.

Altre mòdul es el de **Gestió d'assignatures** que permet a l'usuari crear, modificar o eliminar una assignatura al sistema. Les assignatures seran úniques i al mòdul **Gestió de cursos** es on



s'assigna a un curs i/o altre, d'aquesta manera evitem tenir la mateixa informació repetida tantes vegades com cursos la imparteixin.

Altres mòduls es el de **Gestió de recursos** que permet a l'usuari crear, modificar o eliminar un recurs del centre de formació. Els recursos són elements que poden ser utilitzats pel personal docent com ara projectors, portàtils, reproductors de DVD, etc. A més també permet gestionar l'acceptació o denegació de les peticions que un docent hagi demanat sobre els recursos existents.

L'últim mòdul es el de **Gestió d'usuaris** que permet a l'usuari crear, modifica o eliminar un usuari de tipus *professor* o de tipus *alumne*. A més podrà gestionar la matriculació a cursos dels usuaris de tipus *alumne* que existeixen al sistema.

L'aplicació ASP.Net (WEB) anomenada **GestColesWEB** consta de 2 mòduls ben diferenciats. Cadascun d'ells està orientat a gestionar unes determinades tasques i com que les gestions que poden realitzar cadascun dels usuaris són ben diferents estan pensats per a que un mòdul sigui utilitzat per l'usuari *professor* i l'altre per l'usuari *alumne*. S'han dissenyat els diferents mòduls de l'aplicació que faran servir els usuaris pensant en les necessitats i funcionalitats de cada tasca. Com que aquestes tasques no depenen del Secretari/a i/o Personal Administratiu s'ha estimat realitzar una aplicació *mobile* per a que els usuaris puguin accedir des de qualsevol lloc i en qualsevol moment. A més s'ha optat per una línia continuïsta sobre els mòduls de l'aplicació Windows per aconseguir una aplicació visual fàcil d'entendre i que automatitzi tots aquests processos.

El mòdul **Gestió Utilitats Alumne** permet a l'usuari de tipus *alumne* crear, modificar o eliminar una petició de Tutoria amb un dels seus professors. A més s'inclou una opció per poder consultar el seu expedient acadèmic on hi ha la informació sobre els cursos que ha cursat i dels que està cursant actualment (assignatures, notes, professors...).

El mòdul **Gestió Utilitats Professor** permet a l'usuari de tipus *professor* crear, modificar o eliminar una petició de Recursos del centre. A més s'inclou una opció per gestionar les peticions de tutoria que rep dels seus alumnes acceptant-les o denegant-les i una altra opció per qualificar les assignatures que imparteix als seus alumnes.

A la figura 9 es mostra un resum dels diferents el tipus d'aplicació, mòduls existents, el tipus d'usuari que fa servir cadascun d'ells i la plataforma o tecnologia sobre la que funcionarà.

Aplicació	Mòdul	Usuaris	Tecnologia / plataforma
GestColes	Gestió de Cursos	Secretari/a i/o Personal Administratiu	Aplicació Windows
GestColes	Gestió de Calendaris	Secretari/a i/o Personal Administratiu	Aplicació Windows
GestColes	Gestió d'Assignatures	Secretari/a i/o Personal Administratiu	Aplicació Windows
GestColes	Gestió de Recursos	Secretari/a i/o Personal Administratiu	Aplicació Windows
GestColes	Gestió d'Usuaris	Secretari/a i/o Personal Administratiu	Aplicació Windows
GestColesWEB	Gestió Utilitats Alumne	Alumnes	Aplicació ASP.NET (WEB)
GestColesWEB	Gestió Utilitats Professor	Professors	Aplicació ASP.NET (WEB)

Figura 9: Mòduls de les aplicacions que formen el projecte

## 2.2.2 Descripció del funcionament

A continuació es mostra com interaccionen els usuaris i els mòduls existents de cada aplicació del projecte quan s'inicia un nou any acadèmic:

### *Per a la aplicació Windows GestColes:*

- **Creació del calendari acadèmic o lectiu (any acadèmic):** El secretari/a o personal administratiu crea el calendari lectiu per al proper curs informant de les Festes i tancaments del centre per l'any acadèmic.
- **Inici de curs acadèmic al centre:** Quan el centre inicia l'any acadèmic, el secretari/a o personal administratiu dona d'alta al sistema els nous cursos. Una vegada el curs està en el sistema, es donen d'alta a la base de dades les assignatures noves que s'impartiran i informen de les que ja no n'hi ha el mateix passarà amb els professors i els recursos. Després es procedeix a relacionar als cursos les assignatures que continuaran (tant les noves com de antigues però actives si cal) i a cada assignatura se li relacionarà un professor que la impartirà i un aula on ho farà.
- **Un alumne es vol matricular d'un o dos cursos:** Un cop arriba un estudiant el secretari li preguntarà el curs o cursos que vol matricular i l'informarà si n'hi ha places o no. Si no hi han places l'alumne no es podrà matricular i haurà d'esperar a que un alumne cancel·li la matrícula o al següent any acadèmic. Si pel contrari n'hi ha places se li preguntarà si es o va ser estudiant del centre. En cas negatiu li obrirà un expedient acadèmic amb les dades personals i de contacte. Després, ja sigui nou o no, procedirà a complimentar l'expedient creant la matrícula del curs o cursos on s'indicaran les assignatures que cursarà (depenent del curs en un mateix any lectiu podrà fer totes o no) i li donarà un password per accedir a l'aplicació ASP.NET (WEB).

### *Per a la aplicació ASP.NET (WEB) GestColesWEB:*

- **Un professor vol accedir via WEB al mòdul de Gestió d'Utilitats del Professor:** El professor accedeix al mòdul de l'aplicació **GestColesWEB** utilitzant el seu DNI/NIF/NIE i el Password que li va proporcionar secretaria quan va ser contractat.
- **Un professor fa una petició d'un recurs:** Mitjançant la opció de petició de recursos del centre que hi ha en el mòdul fa un petició d'un o varis recursos. Paral·lelament i de manera asincrònica el secretari/a i/o personal administratiu gestionarà les peticions de recursos acceptant-les o denegant-les des del mòdul **Gestió Recursos** de l'aplicació **GestColes**.
- **Un professor introdueix les notes a un alumne:** mitjançant l'opció de Qualificar a un Alumne accedeix a l'expedient d'un alumne i introdueix o modifica les notes d'aquelles assignatures que l'imparteix.
- **Un alumne vol accedir via WEB al mòdul de Gestió d'Utilitats de l'Alumne:** accedeix al mòdul de l'aplicació **GestColesWEB** utilitzant el seu DNI/NIF/NIE i el Password que li va proporcionar secretaria quan es va matricular.
- **Un alumne vol veure el seu expedient:** mitjançant l'opció de visualització d'expedient se li mostrarà la seva vida acadèmica des de la primera matrícula. Al consultar cada assignatura li apareixerà també la nota que ha tret. També pot veure el calendari lectiu de cada curs on es mostrarà informació sobre les dates d'interès que contingui, etc.

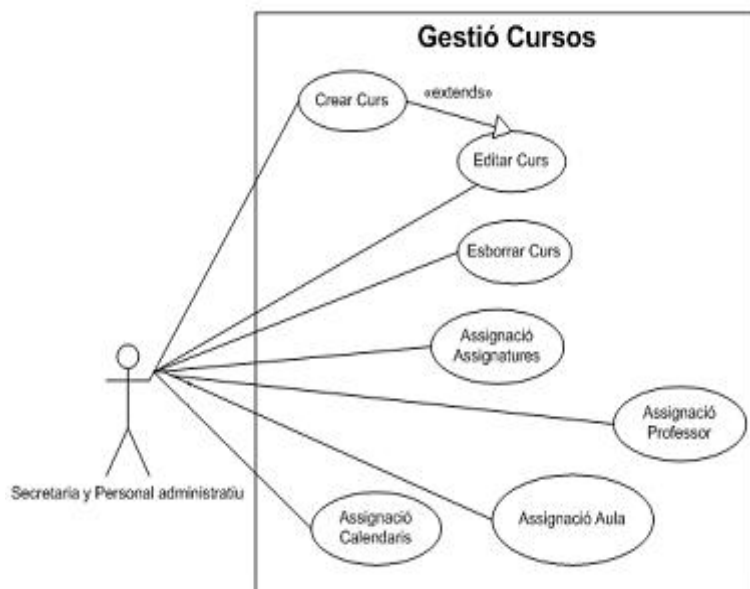
- **Un alumne fa una petició de tutoria a un professor** Mitjançant la opció de petició de tutoria que hi ha en el mòdul fa un petició a un professor. Paral·lelament i de manera asíncrona el professor gestionarà les peticions de tutories acceptant-les o denegant-les des del mòdul **Gestió Tutories** de l'aplicació **GestColesWEB**.

## 2.3 Requeriments funcionals

---

Mitjançant la descripció del funcionament que hem tractat a l'apartat anterior es pot deduir una sèrie de requeriments funcionals que resumirem en forma de diagrama de casos d'ús per a cada mòdul de les aplicacions:

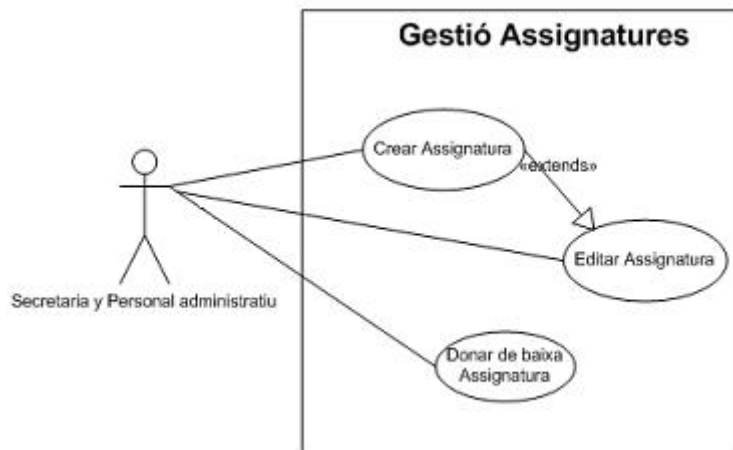
- **Aplicació Windows. Mòdul Gestió Cursos**



Utilitzat pel Secretari/a i personal administratiu. Permet donar d'alta, editar o eliminar un curs al sistema. Si es selecciona un curs de la llista se li podrà assignar un calendari lectiu existent, assignar les assignatures que contindrà, assignar els professors que impartiran aquestes assignatures i assignar les aules on s'impartiran cadascuna d'elles. Son casos d'ús utilitzats eventualment. Cal dir que per les assignacions es guardarà internament les claus identificatives de les dues entitats a relacionar. L'eliminació d'un curs es farà quan no tingui cap assignació relacionada, en cas contrari no es podrà eliminar.

Figura 10: Cas d'Ús Mòdul Gestió Cursos

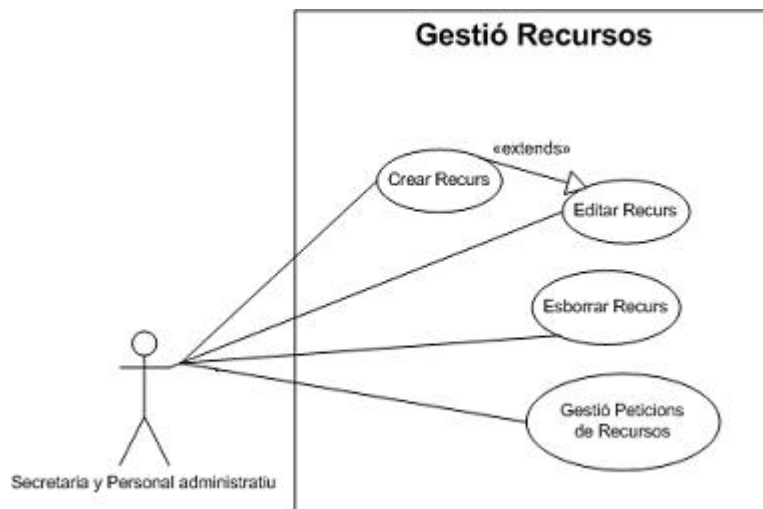
- **Aplicació Windows. Mòdul Gestió Assignatures**



Utilitzat pel Secretari/a i personal administratiu. Permet donar d'alta, editar o eliminar una assignatura al sistema. L'eliminació d'una assignatura es farà quan no tingui cap assignació relacionada, en cas contrari no es podrà eliminar.

Figura 11: Cas d'Ús Mòdul Gestió Assignatures

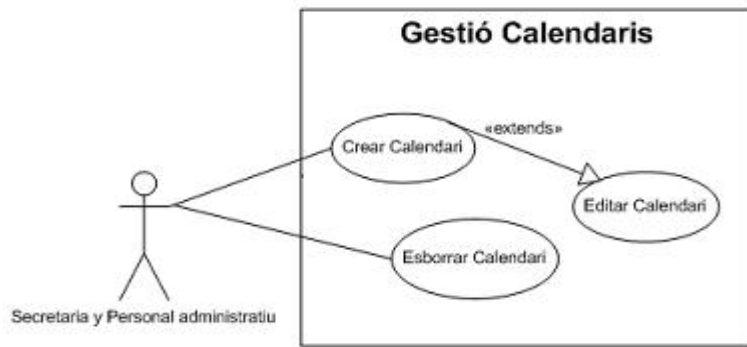
- **Aplicació Windows. Mòdul Gestió Recursos**



Utilitzat pel Secretari/a i personal administratiu. Permet donar d'alta, editar o eliminar un recurs al sistema. L'eliminació d'un recurs es farà quan no tingui cap petició relacionada, en cas contrari no es podrà eliminar per tal de conservar l'històric. També permet gestionar les peticions que un professor realitza sobre els recursos existents.

Figura 12: Cas d'Ús Mòdul Gestió Recursos

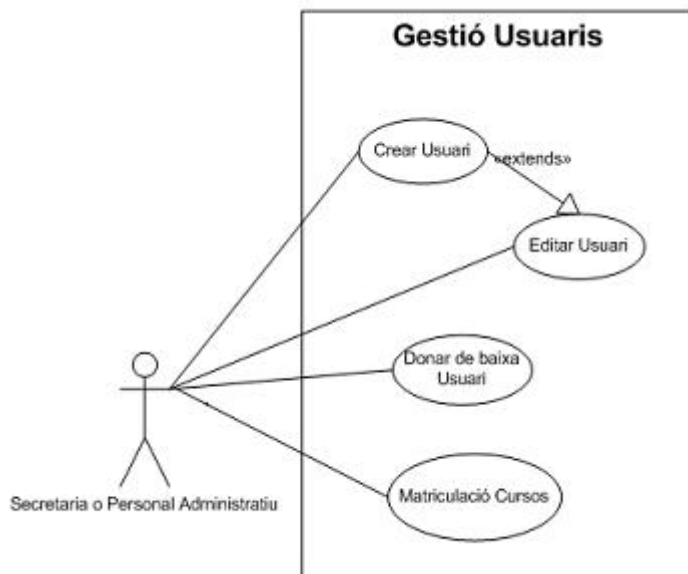
- **Aplicació Windows. Mòdul Gestió Calendaris**



Utilitzat pel Secretari/a i personal administratiu. Permet donar d'alta, editar o eliminar un calendari lectiu al sistema. L'eliminació d'un calendari lectiu es farà quan no estigui assignat a cap curs, en cas contrari no es podrà eliminar.

Figura 13: Cas d'Ús Mòdul Gestió Calendaris

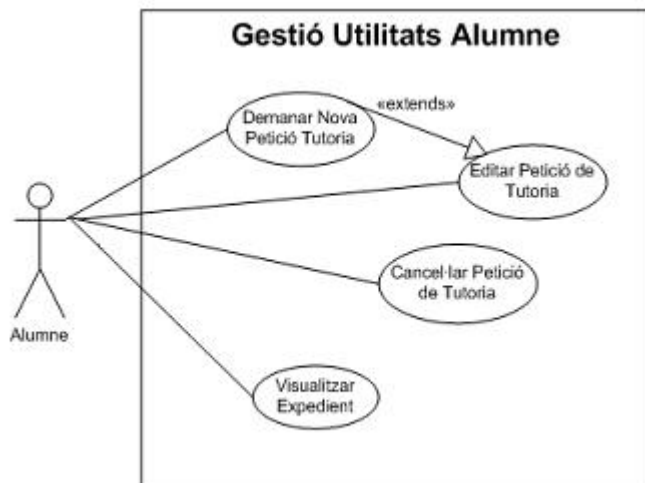
- **Aplicació Windows. Mòdul Gestió d'Usuaris**



Utilitzat pel Secretari/a i personal administratiu. Permet donar d'alta, editar o eliminar un usuari de tipus Alumne o Professor al sistema. L'eliminació d'un usuari es farà quan no estigui o hagi estat matriculat a cap curs, en cas contrari no es podrà eliminar. També podrà fer la matrícula a un usuari seleccionat on informará del curs – calendari que es matricula i les assignatures que cursarà.

Figura 14: Cas d'Ús Mòdul Gestió d'Usuaris

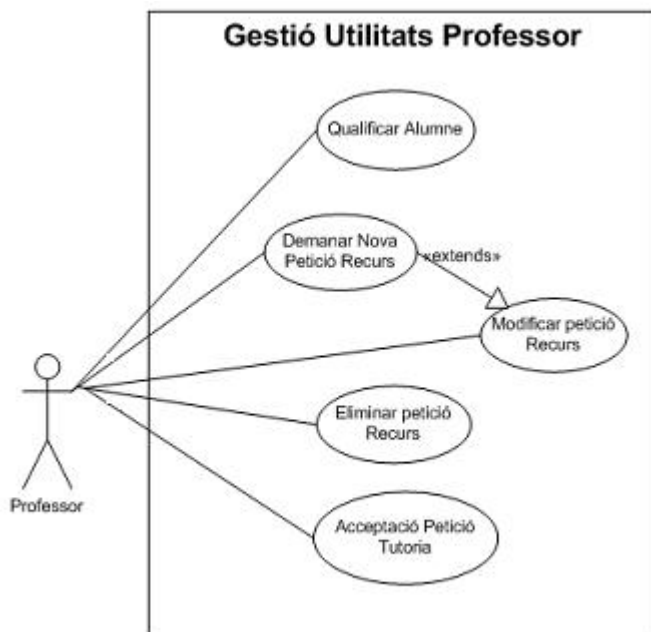
- **Aplicació ASP.NET (WEB). Mòdul Gestió Utilitats Alumne.**



Utilitzat per l'usuari Alumne. Permet donar d'alta, editar o cancel·lar una petició de tutoria a un professor. Només podrà modificar o cancel·lar aquelles peticions que no estiguin aprovades o cancel·lades i que siguin posteriors al moment de voler fer-ho. També té l'opció de visionar el seu expedient acadèmic on veurà tota la seva història acadèmica fins al moment.

Figura 15: Cas d'Ús Mòdul Gestió Utilitats Alumne.

- **Aplicació ASP.NET (WEB). Mòdul Gestió Utilitats Professor.**



Utilitzat per l'usuari Professor. Permet donar d'alta, editar o cancel·lar una petició d'un recurs del centre. Només podrà modificar o cancel·lar aquelles peticions que no estiguin aprovades o cancel·lades i que siguin posteriors al moment de voler fer-ho. També té l'opció de gestionar les peticions de tutoria que rep de l'alumnat i que no estiguin Cancel·lades o siguin de dates anteriors a l'actual. Altre opció es poder qualificar als alumnes als quals el professor imparteix una assignatura.

Figura 16: Cas d'Ús Mòdul Gestió Utilitats Professor.

## 2.4 Escenari del projecte

---

L'escenari on s'utilitzarà aquesta aplicació es un centre de formació on hi haurà un PC servidor amb SQL Server 2000 instal·lat i que contindrà la base de dades del centre. També hi haurà un altre PC servidor WEB amb Internet Information Server (IIS). Una altra possibilitat seria utilitzar el mateix servidor SQL com a servidor WEB però per qüestions de seguretat millor tenir-los separats.

A secretaria hi haurà un o mes PC's on hi hauran instal·lats l'aplicació Windows **GestColes** d'aquest projecte. Com que aquests ordinadors els faran servir el secretari/a o personal administratiu tots tindran instal·lats els 5 mòduls que poden utilitzar.

El sistema disposarà de sortida a Internet mitjançant un router perquè els internautes es puguin connectar a la WEB del centre i així accedir a l'aplicació ASP.NET (WEB) **GestColesWEB**.

Vegem a la figura següent el escenari descrit:

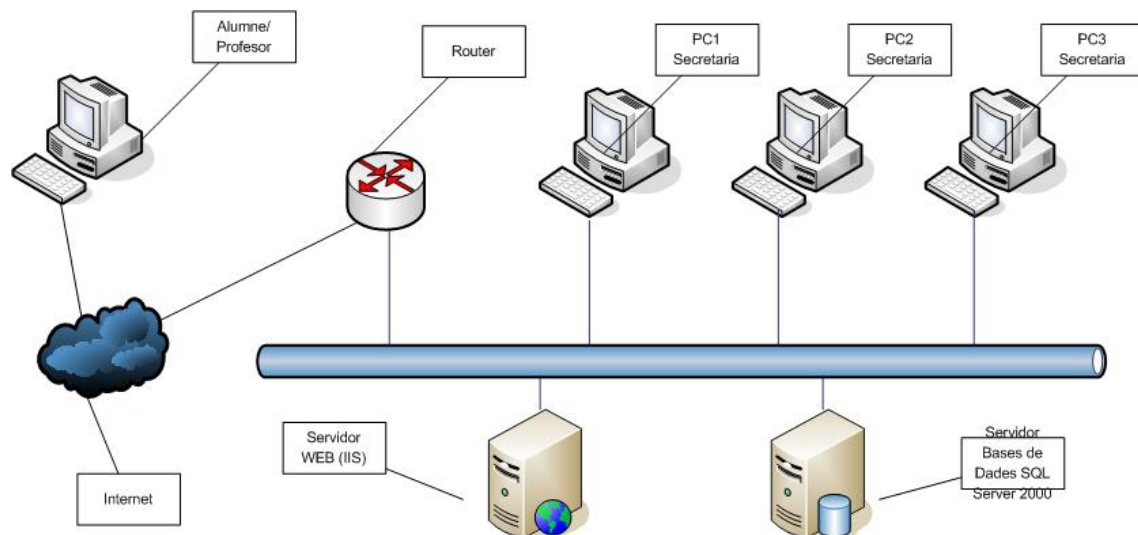


Figura 17: Escenari d'utilització del projecte

## 2.5 Arquitectura de les aplicacions del projecte

---

### 2.5.1 Aplicació Windows

L'arquitectura que farà servir el projecte serà del tipus **client-servidor** que consistirà en una o més aplicacions instal·lades als clients comunicant-se amb la base de dades instal·lada al servidor.

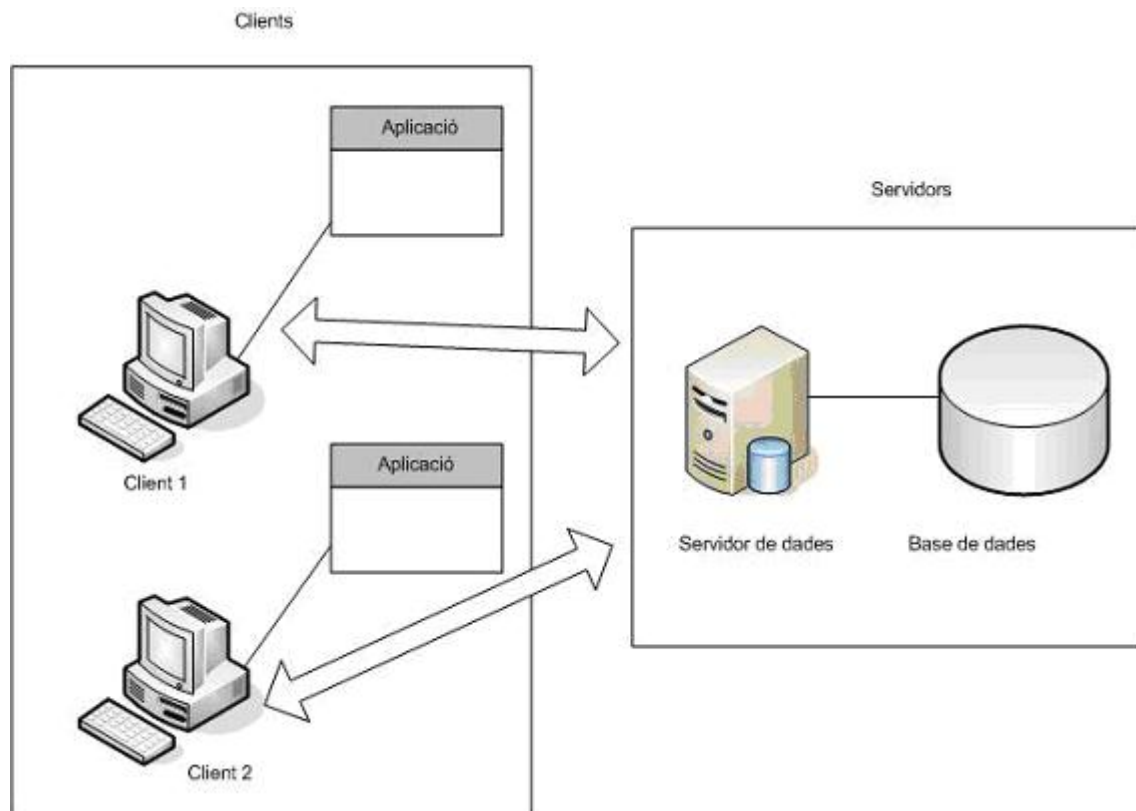


Figura 18: Arquitectura Client - Servidor de l'aplicació Windows del projecte

Amb aquest model, varis clients podrien intentar accedir a les mateixes dades simultàniament per modificar-les creant un problema de concurrència. Però, en el nostre cas, aquesta situació és difícil que es produeixi ja només els faran servir des de la sala de secretaria i encara que accedeixin a la mateixa vegada, la concurrència es petita ja que son pocs usuaris.

De totes maneres, el gestor de base de dades SQL Server bloqueja (locking) un registre que està essent editat per un client de manera que la resta no poden actualitzar la mateixa informació fins que el primer ha acabat.



## 2.5.2 Aplicació WEB

Per als clients que es connectaran a la WEB del centre de formació es necessita un servidor WEB amb IIS (Internet Information Server) instal·lat. El IIS proporcionarà el protocol mitjançant el qual els ordinadors clients es podran connectar al servidor i intercanviar dades. L'aplicació ASP.NET es comunica amb el servidor de base de dades SQL Server 2000 i genera les pàgines necessàries en resposta a les peticions que fan els usuaris WEB.

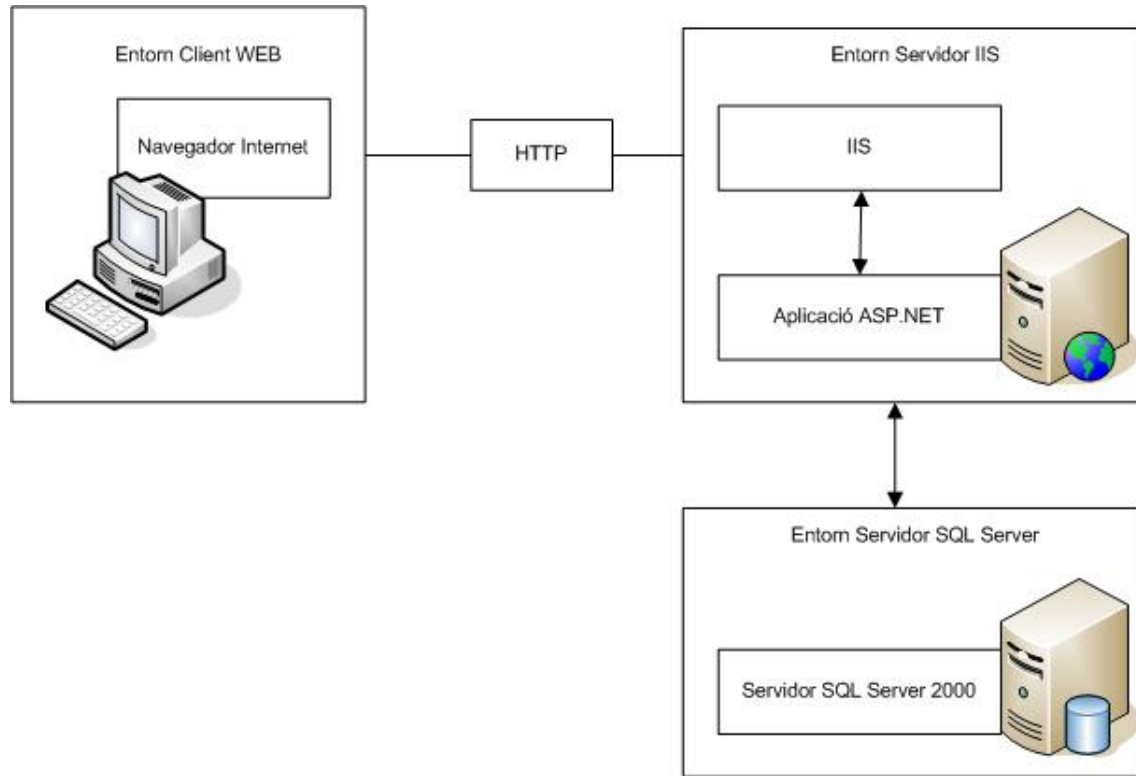


Figura 19: Arquitectura Client - Servidor de l'aplicació WEB del projecte

## 2.6 Disseny de la base de dades

A continuació descompondrem el procés del disseny de la base de dades en 2 etapes:

La primera etapa serà el disseny conceptual on obtenim una estructura de la informació que s'emmagatzema independentment del sistema gestor de base de dades que utilitzem. Aquest apartat ens permet fixar-nos en l'estructuració de la informació sense haver de tenir en compte cap qüestió tecnològica. En aquesta etapa farem servir el model **entitat - relació** abreviar com **ER** per expressar el resultat.

La segona etapa és el disseny lògic. Partint del disseny conceptual de l'etapa anterior, generem un model que es pugui adaptar a la tecnologia que farem servir per implementar la base de dades, en aquest cas, un SGBD relacional com SQL Server 2000.

### 2.6.1 Disseny conceptual

El model ER dissenyat per al projecte conté les següents entitats, atributs i interrelacions:

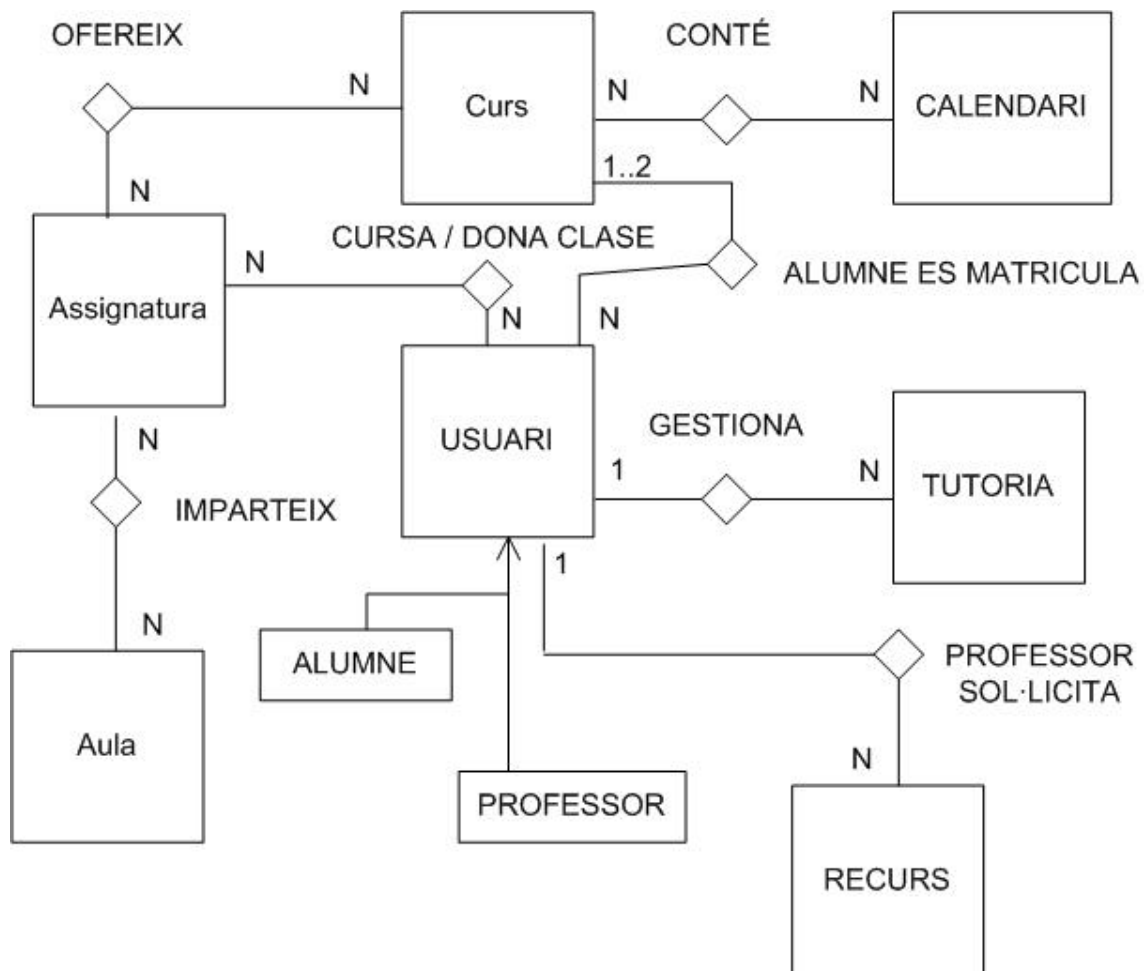


Figura 20: Model ER de la base de dades del projecte

## 2.6.2 Disseny lògic

Partint del model **ER** dissenyat a l'apartat anterior fem la transformació al model relacional obtenint les següents relacions:

### ENTITATS

**tUsuaris**(CodiUsuari, sNom, sCognom1, sCognom2, sSexe, dDataNaixement, sAdresa, sCP, sPoblacio, sProvincia, sTelefon, sTitulacio, dDataAlta, dDataBaixa, sPerfil, sPassword)

**tCursos**(CodiCurs, sNom, sDescripcio, nDuracio, nPreu, dDataBaixa)

**tAssignatures**(CodiAssignatura, sNom, sDescripcio, dDataBaixa)

**tAules**(CodiAula, sNom, bDisponibilitat)

**tRecursos**(CodiRecurs, sNom, sModel, sDescripcio, sEstat)

**tCalendaris**(CodiCalendari, sNom, sDescripcio, sDatesInteressants, dDataInici, dDataFi)

### INTERRELACIONS

**tPeticioRecursos**(CodiUsuari, CodiRecurs, dDataUs, sEstat)

on {CodiUsuari} referencia TUSUARIS

i {CodiRecurs} referencia TRECursos

**tPeticioTutories**(CodiAlumne, CodiProfessor, dDataTutoria, sObservacions, sEstat)

on {CodiAlumne} referencia TUSUARIS

i {CodiProfessor} referencia TUSUARIS

**tMatriculacióAlumneCurs**(id, CodiUsuari, idCalendariCurs)

on {CodiUsuari} referencia TUSUARIS

i {idCalendariCurs} referencia TCALENDARICURSOS

**tMatriculacióAlumneAssignatura**(idMatriculaCurs, CodiAssignatura, nQualificacio)

on { idMatriculaCurs} referencia TMATRICULACIOALUMNECURS

i {CodiAssignatura} referencia TASSIGNATURES

**tImparticioAssignatura**(CodiUsuari, CodiAssignatura, idCalendariCursos)

on {CodiUsuari} referencia TUSUARIS

i {CodiAssignatura} referencia TASSIGNATURES

i {CodiCalendariCursos} referencia TCALENDARICURSOS

**tCalendariCursos**(id, CodiCurs, CodiCalendari)

on {CodiCurs} referencia TCURSOS

i {CodiCalendari} referencia TCALENDARIS

**tCursAssignatura**(id, idCalendariCursos, CodiAssignatura)

on {idCalendariCursos} referencia TCALENDARICURSOS

i {CodiAssignatura} referencia TASSIGNATURES

**tAssignaturaAules**(CodiAula, idCursAssignatura, sHorari)

on {CodiAula} referencia TAULES

i {idAssignaturaCurs} referencia TCURSASSIGNATURES

Figura 21: Transformació del model ER en relacions

**Aclariments:**

Hi haurà que tenir en compte les següents restriccions de domini:

- Un alumne no es pot matricular o cursar més de 2 cursos a l'hora.
- Un recurs no es pot demanar per més d'un professor per a un mateix dia/hora.
- Una tutoria no es pot demanar per a un dia/hora determinat si ja existeix una petició (ja sigui d'altre alumne o ell mateix)
- En un Aula no es pot impartir a una mateixa hora més d'una assignatura.
- No es podrà modificar les qualificacions una vegada s'ha terminat el curs.
- Un curs pot estar en mes d'un calendari si això implica que també es cursarà en una altre data.

A partir de les diferents relacions existents podem fer un diagrama on apareixen detallades aquestes relacions:

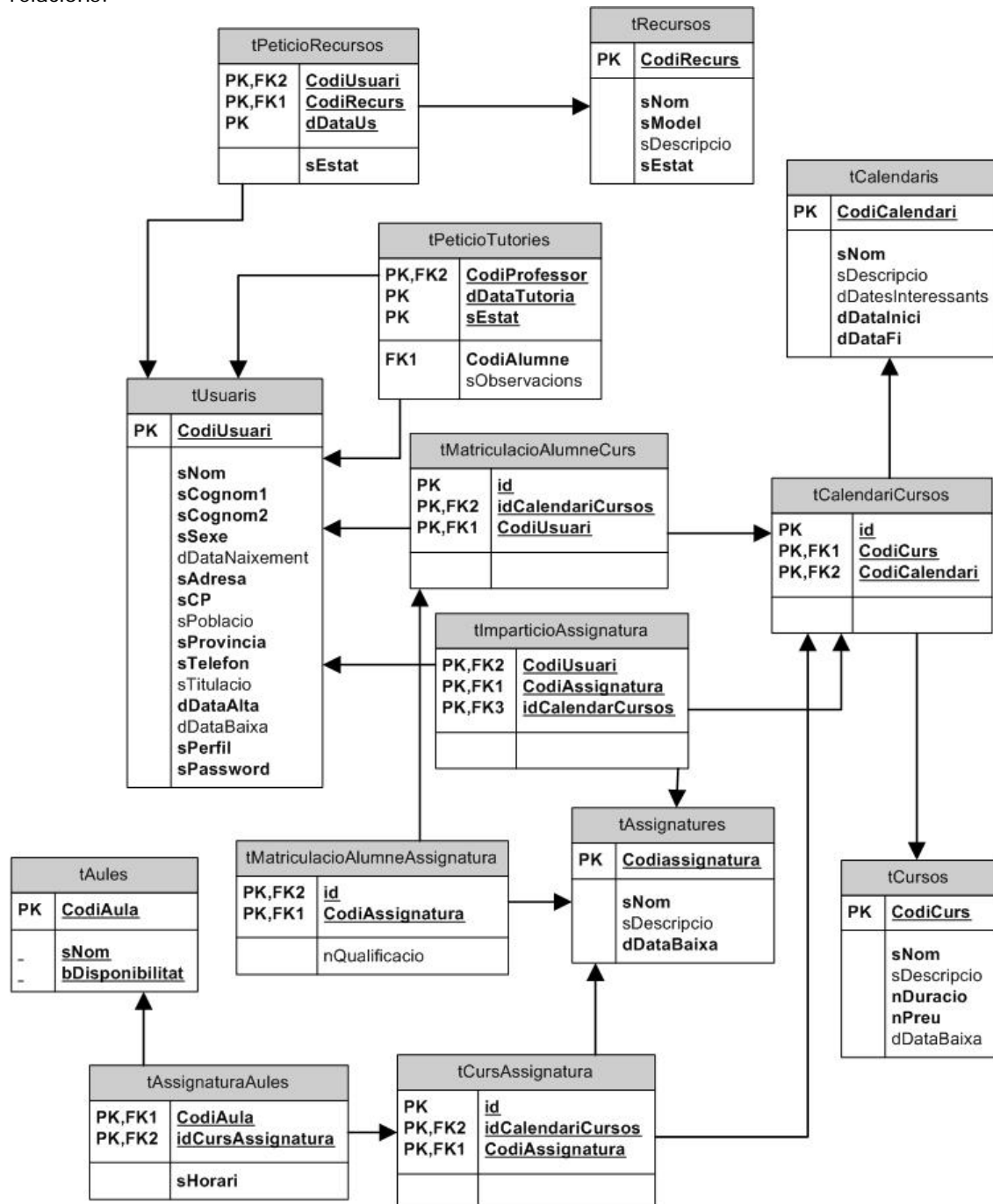


Figura 22: Diagrama lògic de la base de dades del projecte

### 2.6.3 Base de dades GestColes

Aquesta base de dades s'utilitza com emmagatzematge de tota la informació del sistema i a la que s'accedeix tant des de l'aplicació Windows **GestColes** com de l'aplicació ASP.NET **GestColesWEB**.

Es una base de dades estandarditzada que es crearà per al projecte.

## 2.7 Disseny d'estructura de capes de les aplicacions

---

Les dues aplicacions del projecte utilitzen la mateixa estructura de capes. De fet ambdós estan contingudes en una solució que també conté les altres capes que esmentem més avall. Les dues aplicacions interactuen amb una base de dades SQL Server 2000 per tractar dades i es per això que es necessari una bona implementació de l'accés i tractament de les dades. La fórmula escollida es una arquitectura en 5 capes. A continuació detallem en què consisteix cadascuna:

El **codi de gestió dels formularis**, en cas de l'aplicació Windows, i dels **frontpages**, de l'aplicació ASP.NET (WEB), constitueixen una capa que s'encarrega de gestionar les peticions que es reben per la interacció de l'usuari i delegar-la, si es dona el cas, a una altra capa.

**Clases Base:** Aquesta capa constitueix un magatzem de classes. Aquí per exemple hi haurà la classe alumne, la classe assignatura, etc. Es on hi estan les entitats de l'aplicació. Es caracteritzen per contenir una sèrie de propietats que les caracteritzen i mètodes per accedir a aquestes propietats.

**Accés a dades:** Aquesta capa constitueix un magatzem de les classes que hi ha a la capa anterior però només conté els mètodes que requereixen accedir a la base de dades per tractar les dades on una classe té influència. També hi ha una classe que no té relació amb cap de les classes de la capa anterior i que serveix per implementar els mètodes de execució dels dispositius per accedir a la base de dades. Es a dir, una classe on el que hi ha són mètodes que reben una sentència SQL o un Stored Procedure i el que fan es executar-lo i retornar el seu resultat sense tenir en compte qui la demanat. Són mètodes Standard d'accés a dades. Aquesta capa en té la capa Clases Base com a dependència.

**Lògica de negoci:** Aquesta capa també conté una sèrie de classes on rebran una sèrie d'informació i aplicarà la lògica necessària per tal de retornar o executar la comanda demanada amb unes determinades característiques com per exemple depenent del valor d'una dada determinar quin mètode d'accés a dades s'executarà per accedir a certa informació o quin constructor es farà servir per crear una nova instància d'una classe. Aquesta capa en té les capes de Clases Base i Accés a Dades com a dependències.

**Fachada:** Aquesta capa es una capa que serveix per gestionar les peticions rebudes des de el *frontPage* o *formulari* de l'aplicació. Amb la petició pot fer diverses coses, des de gestionar-la ella mateixa com delegar-la a la classe corresponent de la Lògica de Negoci, de l'Accés a Dades o de Clases Base que s'hagi d'encarregar de gestionar-la. Aquesta capa es força important i té les capes Clases Base, Accés a Dades, Lògica de Negoci com a dependències.

A continuació veiem un diagrama amb les dependències de les que parlem

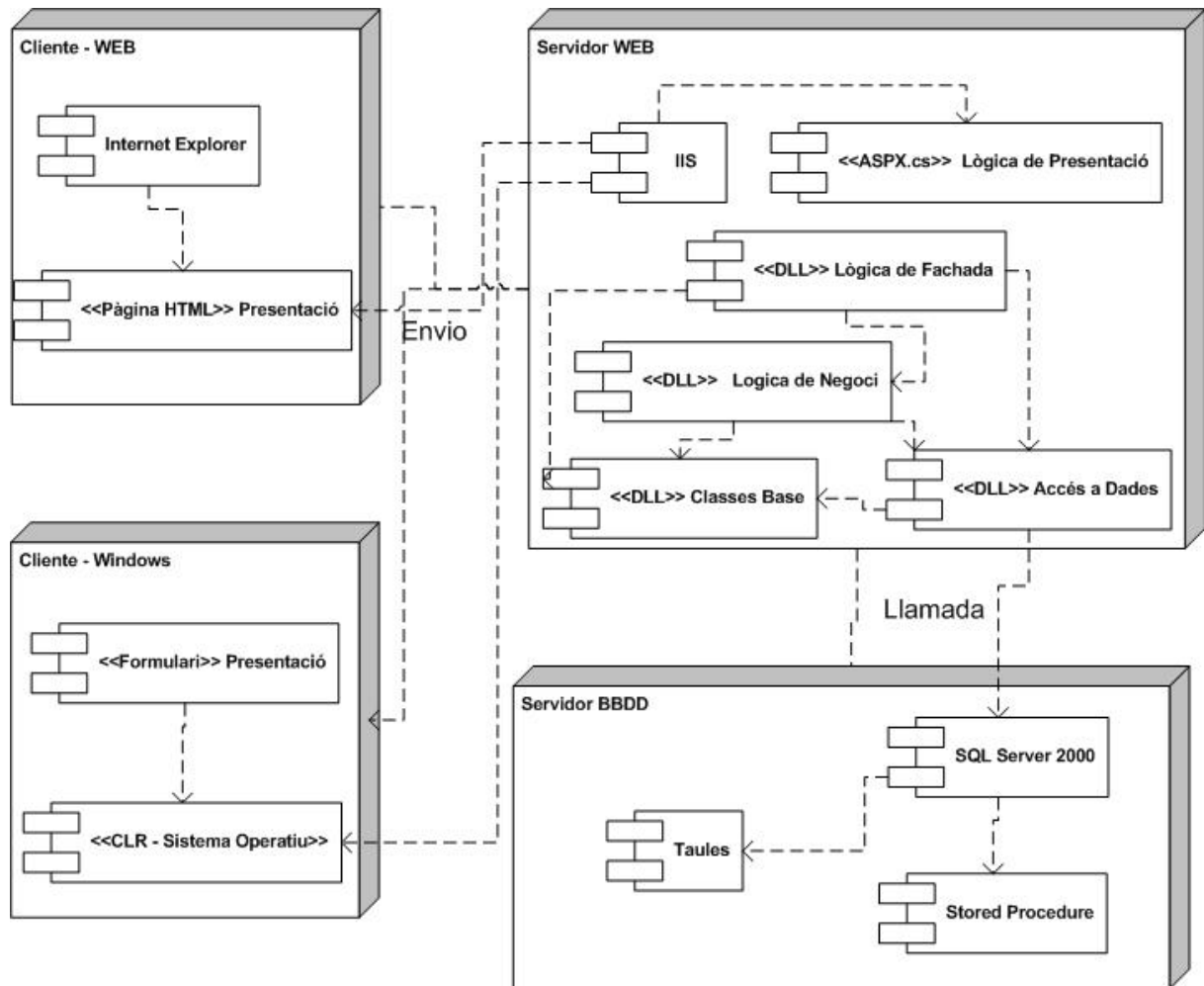


Figura 23: Esquema de l'estructura de capes del projecte

## 2.8 Disseny de classes de les aplicacions

Les classes que fa servir el projecte en les dues aplicacions provenen d'uns projectes de biblioteca de classes agregats a la solució del TFC sota la lògica per capes comentada abans. Per tant les aplicacions no disposen de classes pròpies si excloem les que s'emprenen per la construcció dels formularis Windows o les pàgines WEB i a més una classe d'utilitats amb les funcions reutilitzables en la aplicació per cada formulari o pàgina. Aquests projectes de biblioteca de classes contenen les classes de l'aplicació separant la part d'atributs de la part d'accés a dades i a més hi ha altre part que s'encarrega de gestionar la lògica de negoci de la classe. Es a dir que per una classe en tenim 3 on cadascuna té una finalitat ben diferent:

Al projecte anomenat *ClassesBase* hi hauran només els atributs de la classe i els mètodes d'obtenir o establir el valor d'un atribut.

Al projecte anomenat *Lògicanegoci* hi hauran només els mètodes de gestió de la informació on es decideix quin mètode d'accés a dades s'ha de utilitzar i a més conté mètodes de tractament de dades (com ara posar cometes simples o canviar un valor booleà per un 1 o un 0).

A continuació detallem el projecte anomenat *AccesoDatos* que es el que conté els mètodes principal e importants per accedir a dades per a cada classe:

### Classe Curs

Mètode	Descripció
<code>DataTable LlistaCursos()</code>	Retorna una taula amb la llista de tots els cursos de la Base de Dades
<code>EliminaCurs(string idCurso)</code>	Elimina el curs <i>idCurso</i>
<code>bool CompruebaVigencia(string idCurso)</code>	Retorna un booleà indicant si el curs <i>idCurso</i> es vigent o no.
<code>GuardaCurs(string pParametros)</code>	Guarda les dades del curs indicades en un string <i>pParametros</i> .
<code>bool CompruebaNombreCurso(string sNombre)</code>	Retorna un booleà indicant si el nom del curs <i>sNombre</i> ja existeix a la Base de Dades.
<code>DataTable LlistaCalendarisCursAssignats(string idCurs)</code>	Retorna una taula amb la llista de Cursos assignats a un calendari.
<code>DataTable LlistaCalendarisCursNoAssignats(string idCurs, string DataActual)</code>	Retorna una taula amb la llista de cursos encara no assignats a un calendari.
<code>DataTable LlistaAssignaturesCursAssignades(string idCursCalendari)</code>	Retorna una taula amb la llista d'assignatures assignades a un curs assignat a un calendari
<code>DataTable LlistaAssignaturesCursNoAssignades(string idCursCalendari, string DataActual)</code>	Retorna una taula d'assignatures encara no assignades a cap curs de cap calendari.
<code>DataTable LlistaCalendarisPerCurs(string idCurs)</code>	Retorna una taula de calendaris en el que es troba un curs.
<code>GuardaAssignacioCursCalendari(string pParametros)</code>	Guarda l'assignació d'un curs a un calendari.
<code>EliminaAssignacioCursCalendari(string pParametros)</code>	Elimina l'assignació d'un curs a un calendari.

GuardaAssignacioCursAssignatura( <i>string</i> pParametros)	Guarda l'assignació d'una assignatura a un curs d'un calendari
EliminaAssignacioCursAssignatura( <i>string</i> pParametros)	Elimina l'assignació d'una assignatura a un curs d'un calendari.
<i>string</i> obtenirCursoCalendario( <i>string</i> idCalendariCurs)	Retorna un string amb el nom del curs i del calendari.
<i>bool</i> ComprobaDesAssignacioCursCalendari( <i>string</i> CodiCurs, <i>string</i> sCalendariNOAssignats)	Retorna un booleà que indica si un curs no està assignat a un calendari.
<i>bool</i> ComprobaDesAssignacioCursAssignatura( <i>string</i> sCalendari, <i>string</i> sAssignaturesNOAssignades)	Retorna un booleà que indica si una assignatura no està assignada a un curs d'un calendari.

Figura 24: Mètodes de la classe *Curs* de la biblioteca de classes *AccesoDatos*

### Classe Assignatura

Mètode	Descripció
<i>DataTable</i> LlistaAssignatures()	Retorna una taula amb la llista de totes les assignatures de la Base de Dades
EliminaAssignatura( <i>string</i> idAssignatura)	Elimina l'assignatura <i>idAssignatura</i>
<i>bool</i> CompruebaVigencia( <i>string</i> idAssignatura)	Retorna un booleà indicant si l'assignatura <i>idAssignatura</i> es vigent o no.
GuardaAssignatura( <i>string</i> pParametros)	Guarda les dades de l'assignatura indicades en un <i>string pParametros</i> .
<i>bool</i> CompruebaNombreAssignatura( <i>string</i> sNombre)	Retorna un booleà indicant si el nom de l'assignatura <i>sNombre</i> ja existeix a la Base de Dades.
<i>DataTable</i> LlistaAssignaturesProfessorNoAssignades( <i>string</i> CodiCurs)	Retorna una taula amb la llista d'assignatures encara no assignades a un professor segons el curs-calendari <i>CodiCurs</i> .
<i>DataTable</i> LlistaAssignaturesProfessorAssignades( <i>string</i> CodiProfessor, <i>string</i> CodiCurs)	Retorna una taula amb la llista d'assignatures assignades a un professor segons el professor <i>CodiProfessor</i> i el curs-calendari <i>CodiCurs</i> .
GuardaAssignacioAssignaturaProfessor( <i>string</i> pParametros)	Guarda l'assignació d'una assignatura a un professor.
EliminaAssignacioProfessorAssignatures( <i>string</i> pParametros)	Elimina l'assignació d'una assignatura a un professor.



DataTable LlistaAssignaturesMatriculades( <b>string</b> CodiAlumne, <b>string</b> CodiCalendariCurs)	Retorna una taula d'assignatures matriculades per un alumne a un calendari-curs.
GuardaAssignacioCursCalendari( <b>string</b> pParametros)	Guarda l'assignació d'una assignatura a un curs d'un calendari
DataTable LlistaAssignaturesNoMatriculades( <b>string</b> CodiAlumne, <b>string</b> CodiCalendariCurs)	Retorna una taula d'assignatures encara no matriculades per un alumne a un calendari-curs.
GuardaMatriculacioAssignaturaAlumne( <b>string</b> pParametros)	Guarda l'assignació d'una assignatura d' un curs d'un calendari amb un alumne
EliminaMatriculacioAssignaturaAlumne( <b>string</b> pParametros)	Elimina l'assignació d'una assignatura d'un curs d'un calendari amb un alumne.
DataTable LlistaAssignaturesAulesNoAssignades( <b>string</b> sAula, <b>string</b> CodiCurs)	Retorna una taula amb la llista d'assignatures encara no assignades a un aula.
DataTable LlistaAssignaturesAulesAssignades( <b>string</b> sAula, <b>string</b> CodiCurs)	Retorna una llista d'assignatures assignades a un aula.

Figura 25: Mètodes de la classe *Assignatura* de la biblioteca de classes *AccesoDatos*

### Classe Calendari

Mètode	Descripció
DataTable LlistaCalendaris()	Retorna una taula amb la llista de tots els calendaris de la Base de Dades
EliminaCalendari( <b>string</b> idCalendari)	Elimina el calendari <i>idCalendari</i>
<b>bool</b> CompruebaVigencia( <b>string</b> idCalendari)	Retorna un booleà indicant si el calendari <i>idCalendari</i> es vigent o no.
GuardaCalendari( <b>string</b> pParametros)	Guarda les dades del calendari indicades en un <b>string</b> <i>pParametros</i> .
CompruebaNombreCalendari( <b>string</b> sNombre, <b>string</b> dDataInici, <b>string</b> dDataFi)	Retorna un booleà indicant si el nom del calendari <i>sNombre</i> ja existeix a la Base de Dades.

Figura 26: Mètodes de la classe *Calendari* de la biblioteca de classes *AccesoDatos*

**Classe Recurs**

Mètode	Descripció
<code>DataTable LlistaRecursos()</code>	Retorna una taula amb la llista de tots els recursos de la Base de Dades
<code>EliminaRecurs(string idRecurso)</code>	Elimina el recurs <i>idRecurso</i>
<code>bool CompruebaVigencia(string idRecurso)</code>	Retorna un booleà indicant si el recurs <i>idRecurso</i> es vigent o no.
<code>GuardaRecurs(string pParametros)</code>	Guarda les dades del recurs indicades en un string <i>pParametros</i> .
<code>bool CompruebaNombreRecurso(string sNombre)</code>	Retorna un booleà indicant si el nom del recurs <i>sNombre</i> ja existeix a la Base de Dades.
<code>DataTable LlistaPetitionsRecursos(string bTodos)</code>	Retorna una taula amb la llista de recursos assignats a un calendari.
<code>GuardaPeticioRecurs(string pParametros)</code>	Retorna una taula amb la llista de recursos encara no assignats a un calendari.

Figura 27: Mètodes de la classe *Recurs* de la biblioteca de classes *AccesoDatos***Classe Usuari**

Mètode	Descripció
<code>DataTable LlistaUsuaris()</code>	Retorna una taula amb la llista de tots els usuaris de la Base de Dades
<code>DataTable LlistaUsuaris(string sPerfil)</code>	Retorna una taula amb la llista de tots els usuaris de la Base de Dades amb un determinat Perfil
<code>EliminaUsuari(string idUsuari)</code>	Elimina el usuari <i>idRecurso</i> de la base de dades
<code>bool CompruebaVigencia(string idUsuari)</code>	Retorna un booleà indicant si el usuari <i>idUsuari</i> es baixa o no.
<code>GuardaUsuari(string pParametros)</code>	Guarda les dades de l'usuari indicades en un string <i>pParametros</i> .
<code>bool CompruebaCodiUsuari(string sCodi)</code>	Retorna un booleà indicant si el DNI/NIF/NIE de l'usuari <i>sCodi</i> ja existeix a la Base de Dades.
<code>DataTable LlistaProfessors()</code>	Retorna una taula amb la llista de tots els professors de la Base de Dades.
<code>DataTable LlistaProfessors(string sCodiAlumne)</code>	Retorna una taula amb la llista dels professors que donen alguna classe a un alumne <i>sCodiAlumne</i> .
<code>string ComprobaUsuari(string sCodiUsuari, string sPassword)</code>	Retorna un string amb el perfil d'un usuari.

<pre>DataTable LlistatPeticionsTutories(<b>string</b> sCodiUsuari, <b>string</b> sTodos)</pre>	Retorna una taula amb la llista de                 peticions de tutoria realitzades                 per un alumne <i>sCodiUsuari</i> indicant si vol llistar totes o                 només les vigents <i>sTodos</i>
<pre>GuardaPeticioTutoria(<b>string</b> pParametros)</pre>	Guarda les dades d'una petició de                 tutoria d'un alumne indicades en                 un <b>string</b> <i>pParametros</i> .

Figura 28: Mètodes de la classe *Usuari* de la biblioteca de classes *AccesoDatos*

### Classe Aula

Mètode	Descripció
<pre>DataTable LlistaAules()</pre>	Retorna una taula amb la llista de                 totes les aules de la Base de                 Dades

Figura 29: Mètodes de la classe *Aula* de la biblioteca de classes *AccesoDatos*

### Classe Combo

Mètode	Descripció
<pre>DataTable LlistaSexes()</pre>	Retorna una taula amb la llista de                 sexes definides a la pròpia classe.
<pre>DataTable LlistaPerfils()</pre>	Retorna una taula amb la llista de                 Perfils definits a la pròpia classe.
<pre>DataTable LlistaCalendarisCursAssignats(<b>string</b> idCurs)</pre>	Retorna una taula amb la llista de                 cursos assignats a Calendaris de                 la Base de Dades.
<pre>DataTable LlistatCalendarisCursNoAssignats(<b>string</b> idCurs, <b>string</b> DataActual)</pre>	Retorna una taula amb la llista de                 cursos encara no assignats a                 Calendaris de la Base de Dades.

Figura 30: Mètodes de la classe *Combo* de la biblioteca de classes *AccesoDatos*

### Classe Comun

<pre>DataTable Ejecuta_Sentencia(<b>string</b> p_sSentencia)</pre>	Executa una sentència SQL                 passada i retorna una taula amb                 els resultats.
<pre>Ejecuta_Procedure(<b>string</b> p_sSentencia)</pre>	Executa un stored procedure de                 SQL passat.
<pre><b>int</b> Ejecuta_ProcedureNonQuery(<b>string</b> p_sSentencia)</pre>	Executa un stored procedure de                 SQL passat i retorna el número                 de files afectades.
<pre><b>int</b> Ejecuta_ProcedureScalar(<b>string</b> p_sSentencia)</pre>	Executa un stored procedure de                 SQL passat i retorna una fila amb                 el valor de la primera fila-columna                 del resultat de l'execució.

Figura 31: Mètodes de la classe *Comun* de la biblioteca de classes *AccesoDatos*

A continuació es mostren els mètodes de la classe que serveix com a recurs a l'aplicació Windows.

### Classe Funciones

<code>LlenaCombos(System.Windows.Forms.ComboBox p_cb, System.Data.DataTable p_dtDatos, bool _bBlanco)</code>	Omple un objecte ComboBox indicat amb els valors d'una taula amb la possibilitat que el primer element sigui un element buit.
<code>LlenaCombos(System.Windows.Forms.ListBox p_lb, System.Data.DataTable p_dtDatos, bool _bBlanco)</code>	Omple un objecte ListBox indicat amb els valors d'una taula amb la possibilitat que el primer element sigui un element buit.

Figura 32: Mètodes de la classe *Funciones* del projecte Windows *GestColes*.

A continuació es mostren els mètodes de la classe que serveix com a recurs a l'aplicació ASP.NET (WEB).

### Classe Funciones

<code>static string ObtenerCadenaConexion()</code>	Retorna un string amb la cadena de connexió a la Base de Dades.
<code>MostrarMensaje(System.Web.UI.Page pPagina, string pMensaje)</code>	Mostra un missatge <i>pMensaje</i> emergent per pantalla.
<code>AbrirVentana(System.Web.UI.Page pPagina, string sUrl, int pAncho, int pAlto, bool pScrollbars, bool pMenu)</code>	Obre una nova finestra.
<code>CerrarVentana(System.Web.UI.Page pPagina)</code>	Tanca la finestra indicada.
<code>LlenaCombos(System.Web.UI.WebControls.DropDownList _ddl, string _sSql, bool _bBlanco)</code>	Omple un objecte DropDownList indicat amb els valors d'una taula obtinguda a partir d'una sentència SQL i amb la possibilitat que el primer element sigui un element buit.
<code>LlenaCombos(System.Web.UI.WebControls.DropDownList _ddl, System.Data.DataTable _dtDatos, bool _bBlanco)</code>	Omple un objecte DropDownList indicat amb els valors d'una taula amb la possibilitat que el primer element sigui un element buit.
<code>bool CompruebaFormatoFecha(string sFecha)</code>	Comproba que el format de la Data sigui correcta.
<code>bool CompruebaFormatoHora(string sHora)</code>	Comproba que el format de l'hora i minuts sigui correcta-

Figura 33: Mètodes de la classe *Funciones* del projecte ASP.NET (WEB) *GestColesWEB*.

## 2.9 Seguretat i accés dels usuaris

---

### 2.9.1 Aplicació Windows

No existeix dins dels propis mòduls Windows cap control d'accés d'usuaris a aquesta aplicació. S'entén que els ordinadors de Secretaria estan connectats a una xarxa local on hi haurà un servidor de domini (pot ser el mateix servidor de base de dades) i tots els usuaris hauran d'iniciar sessió Windows autenticant-se en aquell moment mitjançant el seu nom d'usuari i contrasenya.

Des de la classe *comun* de la biblioteca de classes *AccesoDatos* integrada a la pròpia aplicació es proporciona l'usuari i password per autenticar-se a la base de dades

### 2.9.2 Aplicació ASP.NET (WEB)

L'aplicació ASP.NET (WEB) conté un formulari d'identificació d'usuaris a la que s'accedeix des de la WEB del centre. Cal dir que per aquest projecte no s'ha inclòs la creació d'una WEB i suposem que el centre de formació ja en té almenys una.

Per la identificació es demana el DNI/NIF/NIE de l'usuari i el password que li van donar a la primera matrícula. Amb això el sistema comprova si es un usuari vàlid i en cas afirmatiu quin tipus d'usuari es (Alumne o Professor) per donar-li accés a una pàgina o altre.

Per tal de protegir el màxim possible aquestes peticions de dades a la Base de Dades s'han fet servir en la mesura del possible Stored Procedures encapsulant els paràmetres per tal de protegir aquesta informació.

A més s'han limitat les caixes de text al màxim de lletres que pot tenir cada camp i es comproven que les dades introduïdes coincideixin amb el tipus esperat.

També es controla la edició d'un registre de la base de dades inhabilitant totes les opcions en pantalla excloent aquelles que incideixin directament amb el tractament d'aquest registre.

## **2.10 Requisits tècnics de maquinari i programari**

---

Per poder implementar aquest projecte en un centre de formació i que sigui completament operatiu necessitem una sèrie de programari i maquinari (software i hardware). A la següent llista podem veure tot el necessari per al correcte funcionament del projecte en un escenari real.

### **Servidor de base de dades**

- Processador Pentium IV o similar amb 512 MB RAM i interfície de xarxa.
- Sistema operatiu Microsoft Windows Server 2003 recomanat o Microsoft Windows XP Professional amb Service Pack 2.
- Microsoft SQL Server 2000
- Microsoft .NET Framework 1.0.

### **Servidor WEB**

- Processador Pentium IV o similar amb 512 MB RAM i interfície de xarxa.
- Sistema operatiu Microsoft Windows Server 2003 recomanat o Microsoft Windows XP Professional amb Service Pack 2.
- Servidor WEB Internet Information Server versió 5.1 o superior.
- Microsoft .NET Framework 1.0.

A més d'aquests equips es necessiten els següents dispositius i infraestructura per a la xarxa Ethernet:

- HUB per connectar tots els ordinadors de secretaria amb la resta del centre.
- Línia d'accés a Internet d'alta velocitat per poder donar servei WEB.
- Router per a permetre la sortida a Internet.
- Cablejat Ethernet necessari per a tot el centre de formació.
- Impressora per poder treure còpies en paper.
- Antivirus i Antispam per protegir d'atacs tants als PCs com als servidors.

## 3.Descripció de la solució

La solució completa es compon de 2 aplicacions dividides en mòduls. La primera aplicació es una aplicació Windows dividida en 5 mòduls ben diferenciats. La segona aplicació es una aplicació Windows dividida en 2 mòduls.

A continuació es mostren les funcionalitats dels diferents mòduls existents.

### 3.1 Aplicació Windows

---

#### 3.1.1 Introducció

Aquesta aplicació estarà instal·lada als ordinadors de secretaria i només tindrà drets d'execució de l'aplicació aquests usuaris. Des de l'aplicació podrà gestionar els diferents mòduls de que hi consta.

#### 3.1.2 Inici

Quan l'usuari executa l'aplicació, el primer que veu es una finestra amb el logotip del centre de formació i a dalt un menú amb les diferents opcions que pot fer: *Gestió Cursos*, *Gestió Assignatures*, *Gestió Calendaris*, *Gestió Recursos*, *Gestió Usuaris*, *Finestres* i *Sortir*.

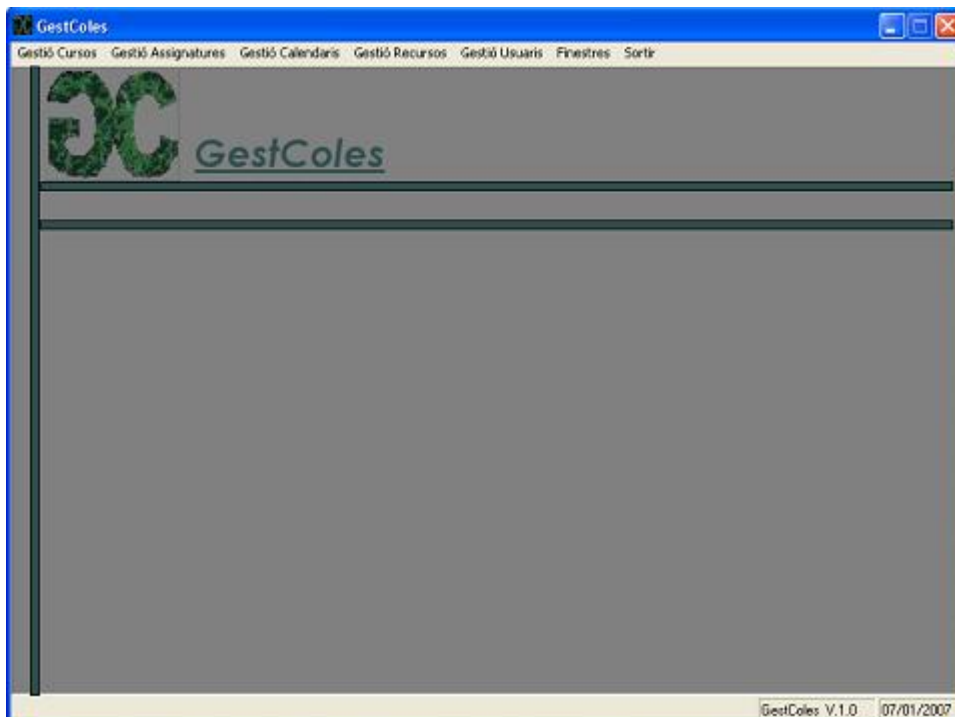


Figura 34: Menú Principal de l'aplicació Windows

### 3.1.3 Gestió Cursos

Aquest mòdul de l'aplicació mostra un llistat de cursos existents al sistema i dona opció a l'usuari a modificar o eliminar-ne algun o de crear un de nou. Si l'usuari selecciona un curs de la llista, les dades del curs apareixen en un formulari de sota i queden inhabilitades totes les opcions excepte les opcions de tractament d'aquestes dades fins que no guardi els canvis o cancel·li l'operació.

A un curs existent l'usuari li pot Assignar Calendaris i/o Assignar Assignatures.

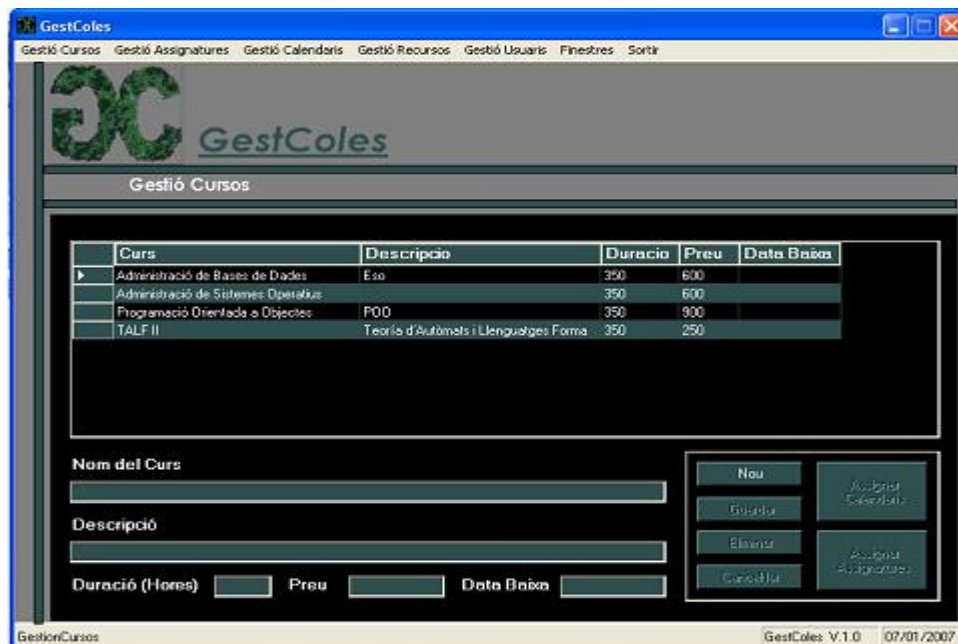


Figura 35: Mòdul Gestió Cursos de l'aplicació Windows

#### 3.1.3.1 Assignar Calendaris

Quan l'usuari des del mòdul anterior selecciona aquesta opció, s'obre una nova finestra on apareix el nom de l'assignatura escollida i dues llistes, una amb els calendaris que ja estan assignats al curs i l'altre amb els calendaris existents no assignats on l'usuari pot assignar i desassignar calendaris.

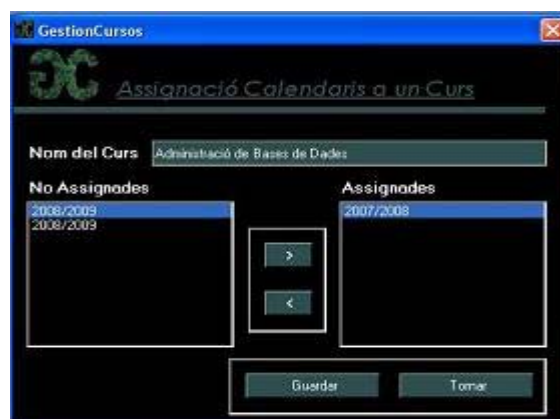


Figura 36: Opció Assignar Calendaris de l'aplicació Windows



### 3.1.3.2 Assignar Assignatures

Quan l'usuari des del mòdul anterior selecciona aquesta opció, s'obre una nova finestra on apareix el nom de l'assignatura escollida, una llista amb els calendaris disponibles i dues llistes, una amb les assignatures que ja estan assignades al curs i l'altre amb les assignatures existents no assignades on l'usuari pot assignar i desassignar assignatures.



Figura 37: Opció Assignar Assignatures de l'aplicació Windows

L'usuari també podrà assignar el professor que impartirà una assignatura i/o l'aula on s'impartirà mitjançant dues opcions a la finestra:

#### 3.1.3.2.1 Assignar professor

Quan l'usuari selecciona aquesta opció apareix una nova finestra on apareix el nom del curs – calendari, una llista per seleccionar professor i dues llistes, una amb les assignatures que ja estan assignades al professor i l'altre amb les assignatures existents no assignades on l'usuari pot assignar i desassignar assignatures. Una assignatura d'un curs – calendari concret només pot ser impartida per un i només un professor.

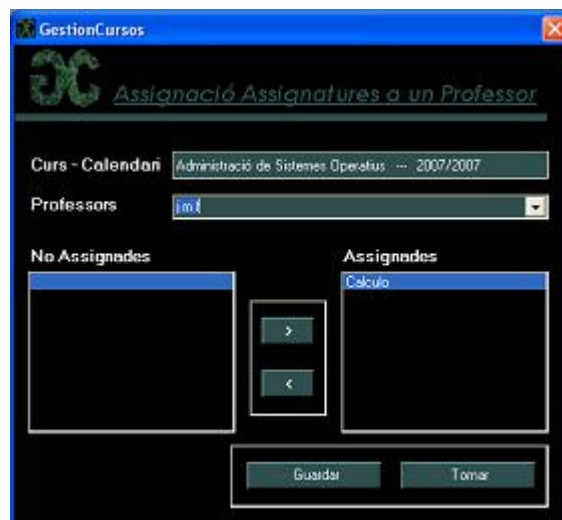


Figura 38: Opció Assignar Professor a unes Assignatures de l'aplicació Windows

### 3.1.3.2 Assignar Aula

Quan l'usuari selecciona aquesta opció apareix una nova finestra on apareix el nom del curs – calendari, una llista per seleccionar aula i dues llistes, una amb les assignatures que ja estan assignades a l'aula de la llista anterior i l'altre amb les assignatures existents no assignades on l'usuari pot assignar i desassignar assignatures. Una assignatura d'un curs – calendari concret només pot ser impartida en una i només una aula.



Figura 39: Opció Assignar Aula a unes Assignatures de l'aplicació Windows

### 3.1.4 Gestió Assignatures

Aquest mòdul de l'aplicació mostra un llistat d'assignatures existents al sistema i dona opció a l'usuari a modificar o eliminar-ne alguna o de crear una de nova. Si l'usuari selecciona una assignatura de la llista, les dades de l'assignatura apareixen en un formulari de sota i queden inhabilitades totes les opcions excepte les opcions de tractament d'aquestes dades fins que no guardi els canvis o cancel·li l'operació.

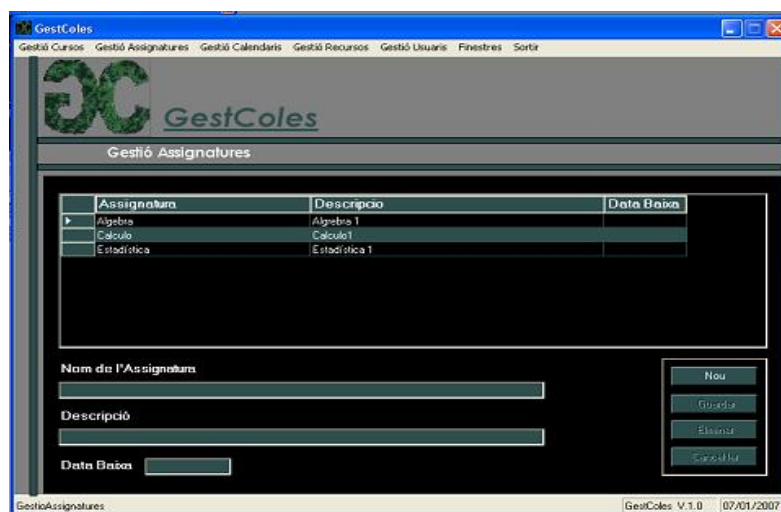


Figura 40: Opció Gestió Assignatures de l'aplicació Windows

### 3.1.5 Gestió Calendaris

Aquest mòdul de l'aplicació mostra un llistat de calendaris existents al sistema i dona opció a l'usuari a modificar o eliminar-ne algun o de crear un de nou. Si l'usuari selecciona un calendari de la llista, les dades del calendari apareixen en un formulari de sota i queden inhabilitades totes les opcions excepte les opcions de tractament d'aquestes dades fins que no guardi els canvis o cancel·li l'operació.

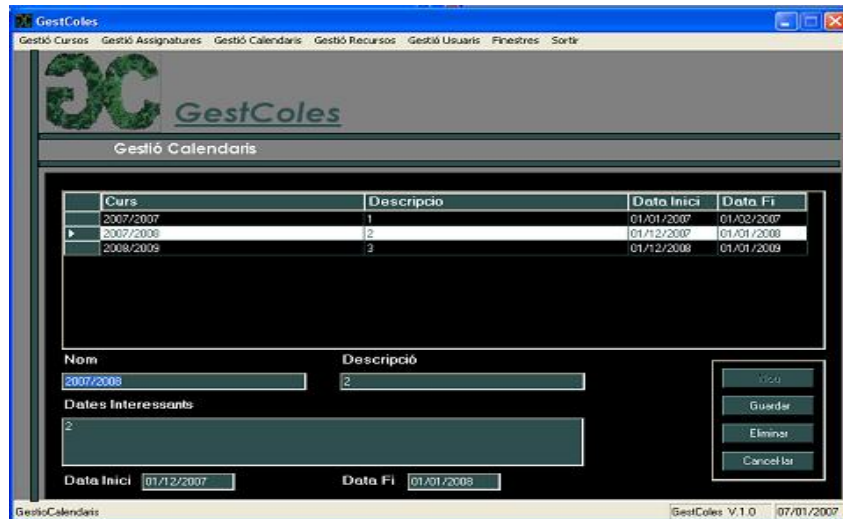


Figura 41: Opció Gestió Calendaris de l'aplicació Windows

### 3.1.6 Gestió Recursos

Aquest mòdul de l'aplicació mostra un llistat de recursos existents al sistema i dona opció a l'usuari a modificar o eliminar-ne algun o de crear un de nou. Si l'usuari selecciona un recurs de la llista, les dades del recurs apareixen en un formulari de sota i queden inhabilitades totes les opcions excepte les opcions de tractament d'aquestes dades fins que no guardi els canvis o cancel·li l'operació. També pot gestionar les peticions de recursos dels professors.

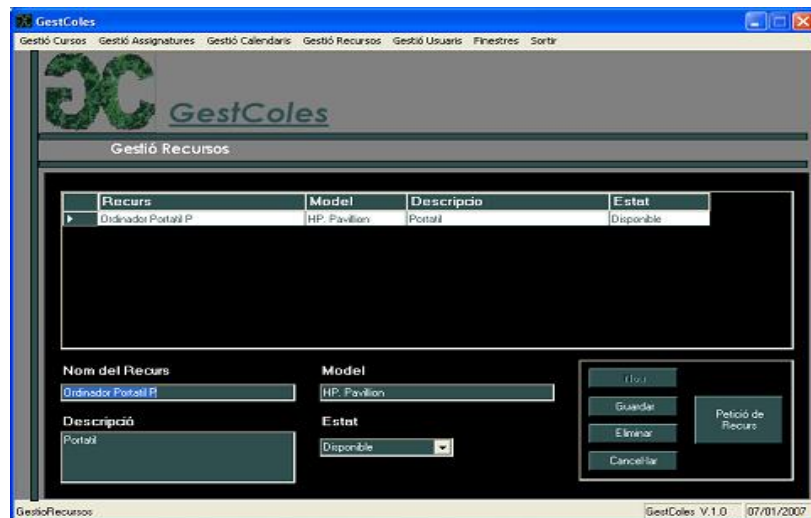


Figura 42: Opció Gestió Recursos de l'aplicació Windows

### 3.1.6.1 Petició de recurs

Quan l'usuari des del mòdul anterior selecciona aquesta opció, s'obre una nova finestra on apareix una llista amb les peticions de recursos per part dels professors i te la possibilitat de veure totes les peticions existents al sistema o només les que estan vigents per confirmar. Si selecciona una petició de la llista, apareix la petició sota la llista i podrà decidir si es acceptada o no.



Figura 43: Opció Assignació Assignatures a un professor de l'aplicació Windows

### 3.1.7 Gestió Usuaris

Aquest mòdul de l'aplicació mostra un llistat dels usuaris existents al sistema i dóna opció a l'usuari a modificar o eliminar-ne algun o de crear un de nou. Si l'usuari selecciona un usuari de la llista, les dades del usuari apareixen en un formulari de sota i queden inhabilitades totes les opcions excepte les opcions de tractament d'aquestes dades fins que no guardi els canvis o cancel·li l'operació. També pot gestionar les matriculacions de l'usuari amb perfil alumne seleccionat.

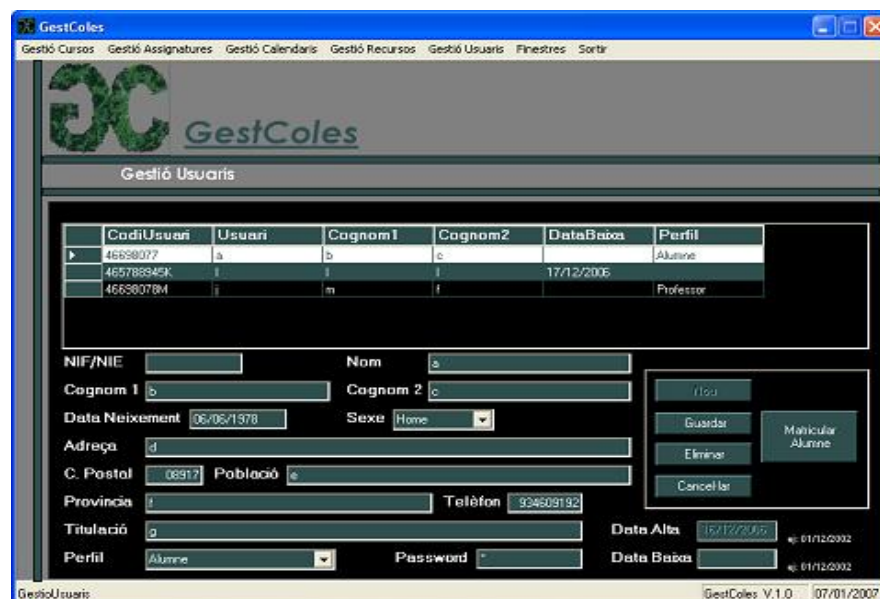


Figura 44: Opció Gestió Usuaris de l'aplicació Windows

### 3.1.7.1 Matricular Alumne

Quan l'usuari des del mòdul anterior selecciona aquesta opció, s'obre una nova finestra on apareix el nom de l'alumne, una llista per seleccionar calendari – curs i dues llistes, una amb les assignatures que ja estan assignades a la relació de l'alumne amb el 0curs - calendari de la llista anterior i l'altre amb les assignatures existents no assignades on l'usuari pot assignar i desassignar assignatures. Un alumne per a un mateix calendari no pot matricular-se en mes de dos cursos.

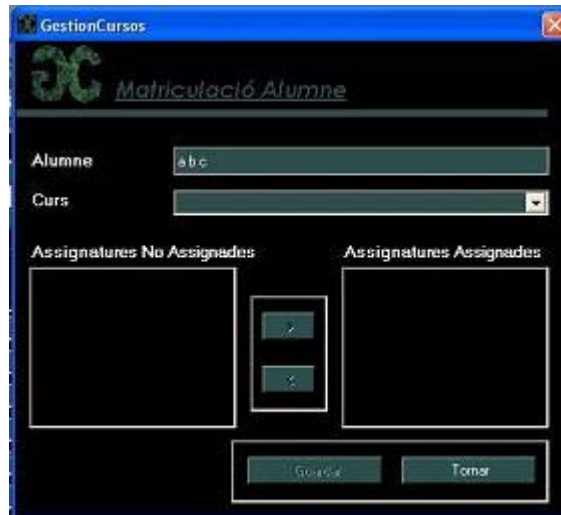


Figura 45: Opció Matriculació Alumne de l'aplicació Windows

### 3.1.8 Finestres

Quan es clicka sobre aquesta opció apareix un llistat de les finestres obertes en aquest moment i al costat del nom de la finestra mostrada actualment apareix un tick per indicar-ho.

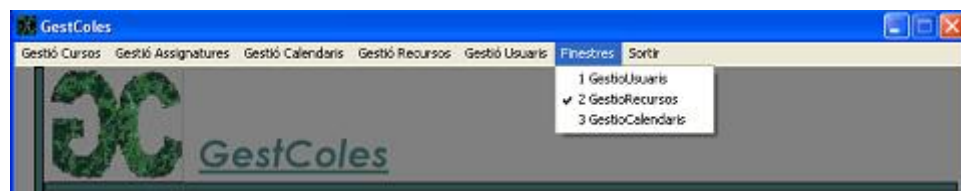


Figura 46: Opció Finestres de menú de l'aplicació Windows

### 3.1.9 Sortir

Quan es clicka sobre aquesta opció, es surt de l'aplicació.

## 3.2 Mòdul WEB

---

### 3.2.1 Introducció

Aquesta aplicació serà accessible a través d'Internet i estarà ubicada a la WEB del centre de formació. Com s'ha dit abans, suposem que el centre de formació té una WEB a Internet i només ens encarregarem de incloure l'accés a l'aplicació ASP.NET (WEB).

### 3.2.2 Inici

Quan l'usuari accedeix a l'aplicació des de la WEB del centre de formació es troba amb una pàgina d'inici de sessió per poder accedir al mòdul de l'aplicació corresponent amb el perfil de l'usuari logat.

### 3.2.3 Inici Sessió

Quan l'usuari accedeix a aquesta pàgina, veurà un formulari per logar-se introduint el seu codi d'usuari (DNI/NIF/NIE) i password. Depenent del perfil de l'usuari logat apareixerà un mòdul o un altre. Si l'usuari té perfil *Alumne* accedirà al mòdul *Gestió Utilitats Alumne*. Si per contra el perfil es de *Professor* accedirà al mòdul *Gestió Utilitats Professor*.

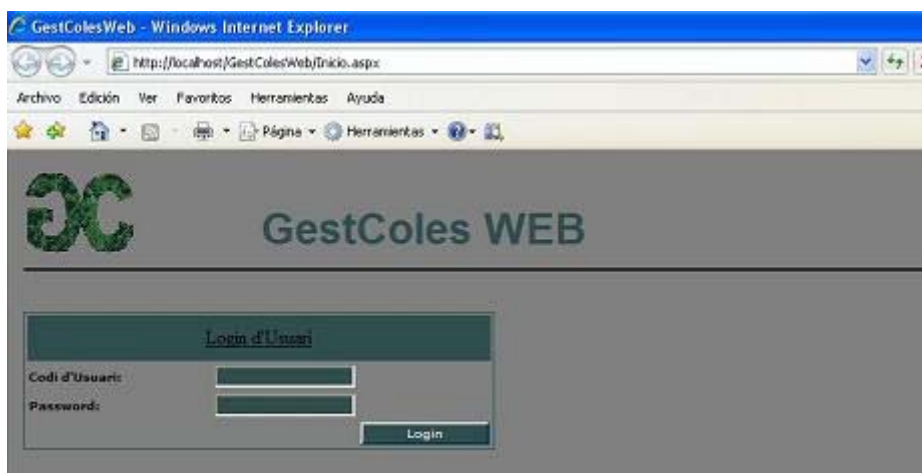


Figura 47: Finestra de Login d'usuari de l'aplicació ASP.NET (WEB)

### 3.2.4 Gestió Utilitats Alumne

#### 3.2.4.1 Petició Tutoria Alumne

Aquest mòdul de l'aplicació mostra un llistat de les peticions de tutoria existents al sistema i dona opció a l'usuari d'escollir si veure totes les peticions o només les vigents pendents de confirmació. Si selecciona una petició vigent pot canviar l'estat entre *Sol·licitada* o *Cancel·lada*. També pot crear una nova petició de tutoria. També hi ha una opció per accedir a una altre finestra on pot veure el seu expedient a mode de consulta.



Figura 48: Finestra de Gestió d'utilitats d'alumne de l'aplicació ASP.NET (WEB)

### 3.2.4.1.1 Visualitzar Expedient

Quan l'usuari accedeix a aquesta pàgina mitjançant la opció del mòdul anterior, veurà un formulari amb les seves dades personals i un llistat de cursos - calendaris en els que està o ha estat matriculat. Al seleccionar un curs - calendari apareixerà un llistat de les assignatures matriculades i la qualificació que hi tenen. Aquesta opció es merament informativa.

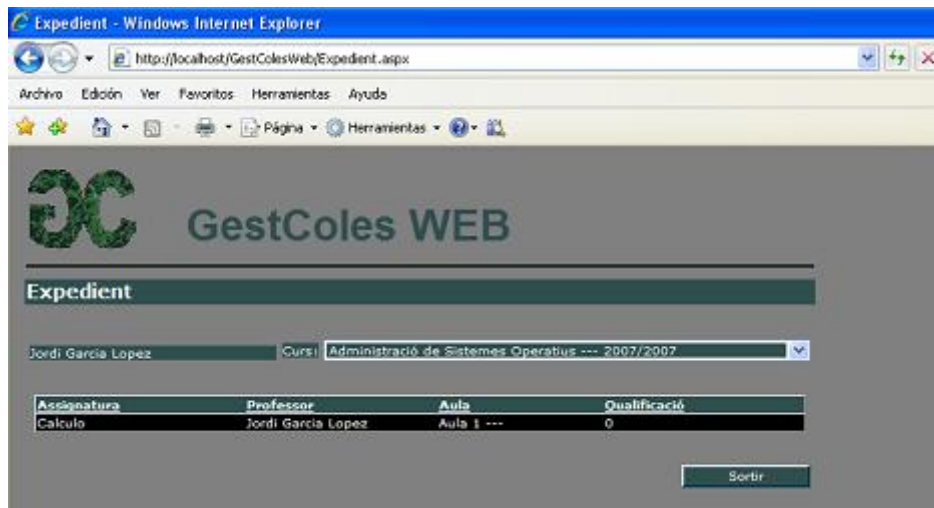


Figura 49: Finestra de l'opció de Visualitzar Expedient de l'aplicació ASP.NET (WEB)



### 3.2.5 Gestió Utilitats Professor

#### 3.2.5.1 Petició Recurs

Aquest mòdul de l'aplicació mostra un llistat de les peticions de recursos existents al sistema i dona opció a l'usuari d'escollir si veure totes les peticions o només les vigents pendents de confirmació. Si selecciona una petició vigent pot canviar l'estat entre *Sol·licitada* o *Cancel·lada*. També pot crear una nova petició de recurs.

També hi ha una opció per accedir a una altre finestra on pot gestionar les peticions de tutoria que li fan els seus alumnes i altre opció per accedir a una finestra on pot qualificar les assignatures dels seus alumnes.

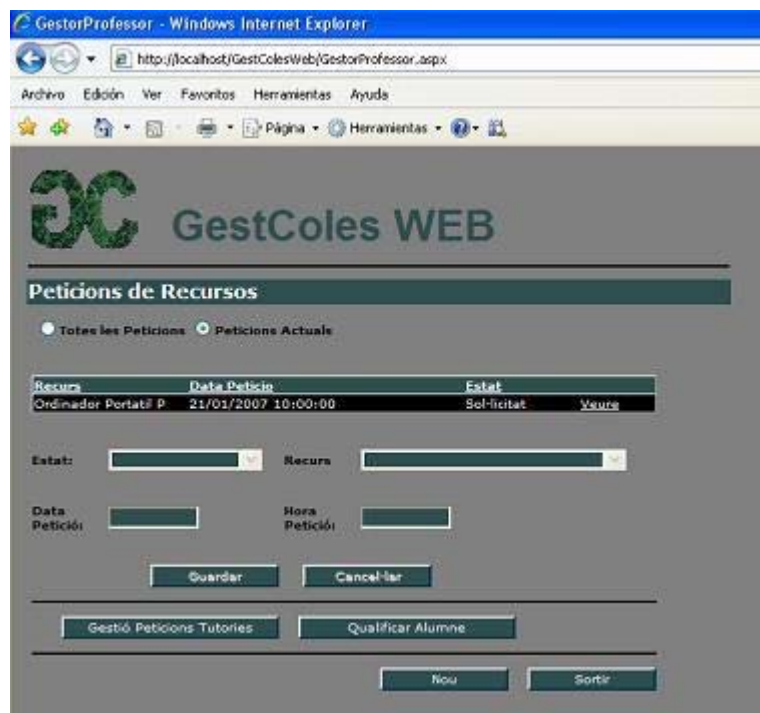


Figura 50: Finestra de Gestió d'utilitats de Professor de l'aplicació ASP.NET (WEB)

### 3.2.5.2 Gestió Peticions Tutories

Aquest mòdul de l'aplicació mostra un llistat de les peticions de tutoria existents al sistema i dóna opció a l'usuari d'escollir si veure totes les peticions o només les vigents pendents de confirmació. Si selecciona una petició vigent pot canviar l'estat entre *Aprovada* o *Cancel·lada*.

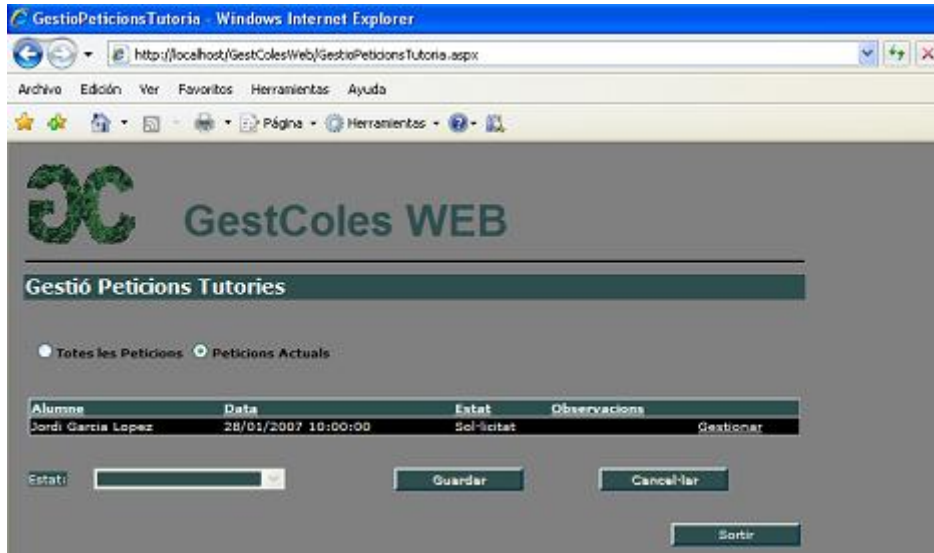


Figura 51: Finestra de l'opció de Gestió Peticions de Tutoria de l'aplicació ASP.NET (WEB)

### 3.2.5.3 Qualificar Alumne

Aquest mòdul de l'aplicació mostra un llistat de la relació de curss – calendari amb les assignatures que imparteix en cadascun i una vegada seleccionada una apareix un llistat dels alumnes que la cursen on pot introduir la qualificació que ha obtingut l'alumne.

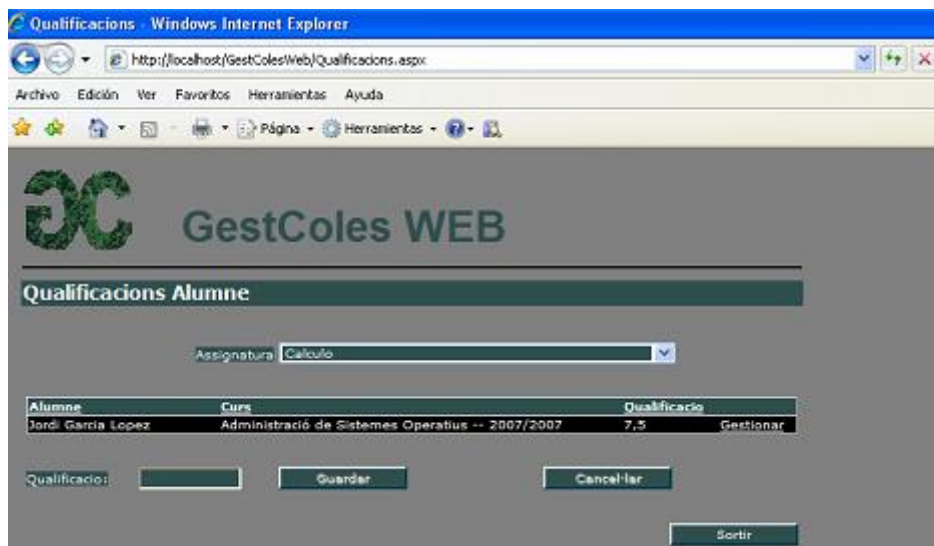


Figura 52: Finestra de l'opció de Qualificar Alumne de l'aplicació ASP.NET (WEB)

## **4.Línies de desenvolupament futur**

És pràcticament impossible que un projecte com aquest que intenta ser una solució completa per a la gestió d'un centre de formació es pugui considerar completament finalitzat. Tot i això la part més important (o el que jo considero més important) i necessària per poder portar el negoci d'un restaurant està contemplada en el producte lliurat.

Resulta evident (i si no ho fos, ja ho dic ara) que jo no pertanyo al ram de la docència ni tinc el més lleuger coneixement sobre com funciona internament un ram com aquest. Així que, segurament, siguin necessàries moltes ampliacions per a que el producte sigui realment competitiu i doni una productivitat alta als seus usuaris.

Realment hauria de preguntar a un expert en la gestió de centres de formació per a que m'assessorés sobre les modificacions o ampliacions que necessita el projecte. Però, entre les que jo puc entreveure estan les següents:

- Incloure un mòdul per a la gestió de les aules que hi pot tenir el centre docent ja que el nombre d'aules entre centres pot variar i a més hi hauria informació si un aula està disponible o no per donar classe.
- Incloure un mòdul de comptabilitat on es podria imprimir les matricules, els rebuts, treure informes econòmics, etc.
- Incloure a l'aplicació WEB un fòrum per que els estudiants i professors puguin comunicar-se i un tauló d'anuncis per poder incloure anuncis d'interès.
- Crear una aplicació per a Pocket PC que contingui les mateixes utilitats que l'aplicació WEB.
- Un sistema d'avis de qualificacions per SMS als mòbils dels alumnes que desitgin aquest servei.

A part d'aquestes ampliacions a nivell de l'aplicació, a nivell de maquinari s'hauria d'incloure un nou servidor per al Pocket PC i que a més disposi de servei d'enviament de SMS.

## 5. Valoració econòmica del projecte

La valoració econòmica que es pot fer del projecte la podem calcular a partir del número d'hores que s'han dedicat a ell. Aquest càlcul serà una mica especial degut a diversos factors:

- No s'ha pogut donar una dedicació exclusiva al projecte si no que s'ha d'haver compartit amb la dedicació a la feina i altres assignatures. Per tant el número emprat d'hores diàries ha estat molt variable.
- L'intentar fer un projecte que englobés la major part de les possibilitats que dona la plataforma .NET (Aplicacions Windows, ASP.NET,...) també ha fet que s'incrementés el número d'hores emprades per a obtenir tots els coneixements necessaris en el desenvolupament del projecte.

Així, els poc mes de 100 dies que s'han dedicat al projecte han suposat unes 450 hores de treball. En aquestes hores s'inclouen les dedicades a totes les etapes del cicle de vida del projecte així com la realització d'aquesta memòria.

El producte final s'ha personalitzat per a un centre de formació imaginari donant-li una imatge final més atractiva. Si bé, les modificacions que caldria realitzar al projecte per a adaptar-lo a un o altre restaurant (incloure les seves imatges, canviar el nom que apareix als formularis de l'aplicació,...) no suposarien una gran despesa de temps i cost. Això es degut a que s'ha intentat durant tot el desenvolupament del projecte que aquests canvis es puguin realitzar de manera senzilla havent de modificar mínimament el codi de les aplicacions.

Adicionalment s'hauria d'afegir al cost del projecte calculat, el cost d'implantació, maquinari i programari (sistemes operatius, servidor de base de dades, etc...) necessaris per al correcte funcionament i posada en marxa en un escenari real. La llista de maquinari i programari addicional que es necessita es pot trobar desglossat a l'apartat **2.10 Requisits tècnics de maquinari i programari**.

## 6. Conclusions

La realització del projecte **Gestió d'un centre de formació amb .NET** ha resultat una experiència enriquidora per un munt de motius.

- El fet d'haver estat capaç de desenvolupar una aplicació completa amb un caire professional a partir d'una tecnologia que es una de les apostes de present i futur més demandades al mercat laboral. El fet de realitzar tot un projecte amb aquest grau de professionalitat m'ha enriquit més en el anàlisi i disseny i donat l'oportunitat de ampliar els coneixements sobre el propi programari.
- Haver après a realitzar una aplicació fent servir un model de treball que intenta aconseguir captar tots els detalls i necessitats que comporta i mantenir un estricte ordre de consecució d'objectius però amb certa flexibilitat que sempre requereix un projecte de mitja - llarga duració.
- Haver aconseguit arribar a tots els objectius fixats al inici del projecte (aquests es poden veure a l'apartat **1.2 Objectius del TFC**).
- Haver pogut posar en pràctica molts dels coneixements adquirits durant aquesta Enginyeria. Sobretot l'aprés en assignatures afines a la programació i a les estructures de dades i de la informació.
- Haver disposat de l'ajuda del consultor a qui li estic agraït per els consells i les explicacions que m'ha donat i han fet possible la consecució del projecte i a la vegada per raonar-los i aplicar-los per en el futur en propers projectes a la vida laboral.

He de dir que m'ha faltat una mica de temps per poder ampliar el projecte i donar-li algunes funcionalitats addicionals per fer-lo un producte més robust i atractiu. Aquestes funcionalitats es comenten a l'apartat **6. Línies de desenvolupament futur**. Tot i això, crec que el producte final obtingut és fiable i suficientment complet.

La valoració final per tant es molt favorable i representa la consolidació i aprofitament de aquests darrers anys que he dedicat a l'estudi d'Enginyeria Tècnica en Informàtica de Sistemes.

## 7. Glossari

**.NET:** La plataforma .NET és una capa de software intermedi que es col·loca entre les aplicacions i el sistema operatiu. La seva finalitat és simplificar el desenvolupament de nou software aportant tots els serveis necessaris per a fer-ho.

**ASP.NET:** Serveis que faciliten la creació d'aplicacions i serveis Web. La darrera versió és la 2.0 que incorpora pàgines mestres, controls per a la navegació, registre i manteniment de la sessió entre d'altres.

**Framework:** (Marc de treball). És una estructura de suport definida en la qual un altre projecte de software pot ser organitzat i desenvolupat. Típicament un framework pot incloure suport de programes, llibreries i un llenguatge de guions entre d'altres programaris per ajudar a desenvolupar i unir els diferents components d'un projecte.

**IDE:** Integrated Development Environment (Entorn de desenvolupament integrat). És un entorn de programació que ha estat empaquetat com si fos un programari d'aplicació, es a dir, consisteix en un editor de codi, un compilador, un depurador i un constructor d'interfície gràfica.

**IIS:** Internet Information Server. (Servidor d'informació d'Internet). IIS és un servidor Web. Això és una sèrie de serveis pels ordinadors que funcionen amb Windows. Aquests serveis converteixen un ordinador en un servidor d'Internet permetent a aquests ordinadors publicar pàgines Web tant local com remotament.

**Model ER:** Entity-relationship model (Model entitat-relació). Model que permet representar el disseny conceptual d'una base de dades independentment del sistema gestor de base de dades que s'utilitzi i sense tenir en compte cap qüestió tecnològica.

**Pocket PC:** És un ordinador de butxaca dissenyat per ocupar el mínim espai i ser fàcilment transportable que executa un sistema operatiu Windows CE de Microsoft que li proporciona capacitats similars a un PC d'escriptori.

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

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

**Grid:** Es un potent component que ens permet crear un entramat de celes. Aquestes celes son altament personalitzables al poder modificar una gran quantitat de propietats, podent fer que siguin des de una simple reixa de informació, fins una complexa aplicació on poder modificar dades dins de les celes.

## **8. Bibliografia i referències**

Per a aquest projecte només s'ha consultat a:

MSDN Library

<http://msdn.microsoft.com/library/default.asp>

MSDN Espanyol.

<http://www.microsoft.com/spanish/msdn/spain/default.mspx>

Buscador Google

<http://www.google.com>

Pàgina Web Personal de " El Guille "

<http://www.elguille.info/>

## 9.Anex. Consideracions de l'aplicació.

### 9.1 Consideracions de les aplicacions del projecte

---

Cal incidir en algunes consideracions que s'han de tenir en compte per treballar amb la aplicació de forma correcta:

- L'aplicació Windows consta d'una finestra principal on apareixen les diferents opcions que donen accessos als mòduls de l'aplicació. Quan s'accedeix a un mòdul, s'obre una nova finestra dins la mateixa finestra principal del programa. En canvi, les opcions a las que es pot accedir des d'un mòdul apareixeran en noves finestres independents i a mode que mentre no es tanqui la finestra principal quedarà inactiva.
- Per controlar les finestres obertes a la finestra principal hi ha una opció anomenada *Finestres* on sortirà una llista de les finestres obertes en aquest moment i marcada amb un tick la finestra activa (que es visualitza) en aquest moment.
- L'aplicació ASP.NET (WEB) consta d'una finestra d'Internet on apareixeran les successives finestres dels diferents mòduls i opcions a mida que es vagi accedint.
- Quan a les aplicacions es selecciona un element d'una llista Grid (ja sigui mitjançant doble click a l'aplicació Windows o amb el botó de la llista Grid per tal efecte) automàticament s'inhabiliten algunes opcions i la pròpia llista Grid per evitar creuaments i distorsió de dades. Amb el botó *Cancel·lar* que hi ha a la finestra s'habiliten altre cop aquestes opcions. En cas d'una edició de l'element seleccionat, els botons de *guardar* i *eliminar*, si n'hi ha, també fan aquesta funció d'habilitació.
- Per accedir a algunes opcions del mòduls pot ser necessari seleccionar prèviament un element d'una llista o d'una llista Grid.
- Quan es crea un curs i s'assigna a un calendari lectiu, es podrà llavors assignar assignatures a aquesta relació curs – calendari, i així a les assignatures assignades a aquests curs – calendaris se li podran assignar un professor (i només un) que la impartirà i un aula (i només una) on s'impartirà.