



Facultad 8

**TRABAJO DE DIPLOMA PARA OPTAR POR EL TÍTULO DE  
INGENIERO EN CIENCIAS INFORMÁTICAS**



*“Sistema de Gestión de Información de la Facultad #8. Módulo  
para la gestión del perfil y los cursos optativos.”*

*Autores: Leonardo San Román Labaut.*

*Reynier Cartaya Hidalgo.*

*Tutora: Ing. Yorgelys González López.*

Ciudad de la Habana, Junio 2007

“Año 49 de la Revolución”

## **DECLARACIÓN DE AUTORÍA**

Declaramos que Reynier Cartaya Hidalgo y Leonardo San Román Labaut somos los únicos autores de este trabajo y autorizamos a la Facultad 8 de la Universidad de las Ciencias Informáticas (UCI) para que haga el uso que estimen pertinente con este trabajo.

Para que así conste firmamos la presente a los        días del mes de Junio del 2007.

\_\_\_\_\_  
Firma del Autor  
Leonardo San Román Labaut

\_\_\_\_\_  
Firma de Autor  
Reynier Cartaya Hidalgo

\_\_\_\_\_  
Firma de la Tutora  
Ing. Yorgelys Gonzalez Lopez.

## AGRADECIMIENTOS

*Queremos agradecer:*

*A nuestra revolución y a Fidel.*

*A la tutora Yorgelys González Pérez que con tanto esfuerzo y dedicación supo guiarnos en el desarrollo de este trabajo.*

*Reynier:*

*A Aranay que sin su apoyo y comprensión en estos 2 años no sé que habría sido de mí, gracias mi niña.*

*A mis abuelos Dora, Eleodoro, Julia y Luis ellos siempre me enseñaron algo nuevo de su vasta experiencia.*

*A mi tío José en especial porque siempre ha estado pendiente de mí como un padre y a su esposa Alicita.*

*A todas mis hermanas, María Julia, Ania, Sandra y Raisa.*

*A todos mis sobrinos.*

*A toda mi familia que no ha tenido reparos en ayudarme a logra este sueño de ser el primer ingeniero.*

*A los amigos de la secundaria y el pre la gente del Calero y el Rancho Nisael, Yamaiji, El Calva, el chino y el Nini.*

*En especial a mis hermanos Lisban y Julio.*

*A mis amigos de la UCI, a mi grupo desde el 1110 hasta el 8501, Roberkis, Javier (Peperon), Jorge, Leonardo, Andy, Karen, Dianeyi, Aylene y todos los demás que fueron muchos.*

*Leonardo:*

*A mis padres que confiaron en mí, quienes me han educado y apoyado en todo lo correcto ensañándome a esforzarme y no amedrentarme ante las dificultades. Quienes dieron con amor todo cuanto pudieron para que lograra este éxito que hoy es de nosotros.*

*A mi hermano que siempre me ha apollado y querido pese a todas las dificultades y la distancia.*

*A mi abuela y a mi tío Juan Antonio que siempre he sido para ellos como un hijo más aquel que no tuvieron pero que han criado y querido como si lo fuera.*

*A mi tío Robertico que me ha brindado todo su Apollo en todo momento.*

*A Rene que siempre ha estado hay apollandome en todo lo que me ha heho falta y brindandome todo su Apollo.*

*A mis Amigos que han compartido conmigo, que me han ayudado y han estado apollandome en todo momento. Por citar algunos, Fabian, Sandy, Ismael, Verdecia, Andy, Reynier, Javier, Jorge, Yoisbel, Marlón, entre otros.*

## DEDICATORIA

*A mis padres, Teresita de Jesús Labaut Sueiro y Asterio San Román Ocha por guiarme por el buen camino, gracias a ellos he obtenido lo logrado....*

*De su hijo Leonardo San Román Labaut.*

*A mis padres, Norma Hidalgo Otaño y Basilio Cartaya Barreto por el empeño y dedicación con que me han conducido en la vida para lograr este resultado....*

*De su hijo Reynier Cartaya Hidalgo*

**PENSAMIENTO**

*“Quien no quiere pensar es un fanático; quien no puede pensar, es un idiota; quien no osa pensar es un cobarde.”*

*Sir Francis.*

## **RESUMEN**

En la UCI<sup>1</sup> fluye información relacionada con los cursos optativos (CO) ofertados a los estudiantes durante la carrera y el perfil como condición obligatoria para la formación integral del graduado, la cual es procesada mediante múltiples vías y por el personal indicado. En la Facultad #8, estos datos se controlan y procesan de manera manual, con el objetivo de obtener estadísticas según necesidades o peticiones de los implicados y directivos, así como para la toma de decisiones, provocando en ocasiones: gasto de tiempo, mano de obra, algunos incumplimientos en la entrega de información, insatisfacciones. Esta investigación pretende elevar el nivel tecnológico de informatización en la facultad para favorecer la gestión y control de la información por el personal autorizado, mediante la implementación del “Sistema de Gestión de Información de la Facultad #8. Módulo para la gestión del perfil y los cursos optativos”, el cual se caracteriza por un eficiente trabajo en red, por ser sencillo, agradable a los ojos del usuario y con un alto grado de rapidez del flujo de datos a procesar. Se pretende además, que se facilite la realización de los procesos relacionados con la matrícula, organización y control de la información de los CO y el perfil, generándose en formato Excel algunos reportes con significativo valor. Este documento recoge todo el proceso de análisis, diseño, implementación y prueba del software desarrollado, donde se modelaron muchos de los artefactos de estos flujos, satisfaciendo los requerimientos del cliente y cumpliendo los objetivos propuestos.

---

<sup>1</sup> Universidad de las Ciencias Informáticas

## ÍNDICE

INTRODUCCIÓN.....	1
CAPÍTULO 1.....	8
FUNDAMENTACIÓN TEÓRICA .....	8
1.1.    Introducción .....	8
1.2.    Estado del arte .....	8
1.2.1.    Conceptos fundamentales.....	8
1.2.2.    Actualidad del trabajo.....	9
1.3.    Tecnologías a utilizar .....	13
1.4.    Arquitectura utilizada .....	19
1.5.    Herramientas utilizadas en la propuesta de solución.....	22
1.6.    Metodologías de desarrollo de software .....	24
1.7.    Conclusiones .....	30
CAPÍTULO 2.....	31
CARACTERÍSTICAS DEL SISTEMA .....	31
2.1.    Introducción .....	31
2.2.    Reglas del negocio.....	31
2.3.    Descripción de los procesos del negocio.....	34
2.4.    Requerimientos funcionales.....	46
2.5.    Requerimientos no funcionales.....	48
2.6.    Definición de los Casos de uso del sistema .....	50
2.7.    Conclusiones .....	72
CAPÍTULO 3.....	73
ANÁLISIS Y DISEÑO DEL SISTEMA.....	73
3.1.    Introducción .....	73
3.2.    Análisis.....	73
3.3.    Diseño.....	78
3.4.    Descripción de las clases.....	85

3.5. Conclusiones .....	92
CAPÍTULO 4.....	93
IMPLEMENTACIÓN Y PRUEBA.....	93
4.1. Introducción .....	93
4.2. Implementación .....	93
4.3. Modelo de prueba.....	102
4.4. Conclusiones .....	106
CONCLUSIONES.....	107
RECOMENDACIONES .....	108
REFERENCIAS BIBLIOGRÁFICAS.....	109
BIBLIOGRAFÍA.....	111
GLOSARIO DE TÉRMINOS.....	113

## **ÍNDICE DE TABLAS**

Tabla 1 Descripción de los actores del negocio. ....	34
Tabla 2 Descripción de los trabajadores del negocio. ....	35
Tabla 3 Descripción textual del CUN Gestionar Cursos Optativos. ....	37
Tabla 4 Descripción textual del CUN Gestionar Perfil. ....	40
Tabla 5 Descripción textual del CUN Solicitar Información. ....	42
Tabla 6 Descripción de los actores del negocio. ....	50
Tabla 7 Descripción textual del CUS Autenticar Usuario. ....	53
Tabla 8 Descripción textual del CUS Gestionar Cursos Optativos. ....	54
Tabla 9 Descripción textual del CUS Gestionar Curso. ....	57
Tabla 10 Descripción textual del CUS Gestionar Perfil. ....	61
Tabla 11 Descripción textual del CUS Gestionar Información. ....	66
Tabla 12 Descripción de clases utilizadas en todos los CUS. ....	85
Tabla 13 Descripción de clases utilizadas en el CUS Gestionar Curso Optativo. ....	88
Tabla 14 Caso de Prueba del CUS Gestionar Curso Optativo: Matricular en un Curso. ....	103
Tabla 15 Caso de Prueba del CUS Gestionar Curso Optativo: Matricular en un Curso. ....	103
Tabla 16 Caso de Prueba del CUS Gestionar Curso Optativo: Matricular en un Curso. ....	104
Tabla 17 Caso de Prueba del CUS Gestionar Curso Optativo: Matricular en un Curso. ....	105
Tabla 18 Caso de Prueba del CUS Gestionar Curso Optativo: Eliminar Curso con nota 0. ....	105

## **ÍNDICE DE FIGURAS**

Fig 1 Arquitectura de 3 capas. ....	21
Fig 2 Fases e Iteraciones de RUP. ....	26
Fig 3 Metodología XP. ....	27
Fig 4 Metodología FDD. ....	28
Fig 5 Diagrama de Casos de Uso del Negocio. ....	36
Fig 6 Diagrama de clases de análisis Autenticar Usuario. ....	73
Fig 7 Diagrama de clases de análisis Gestionar Cursos. ....	74
Fig 8 Diagrama de clases de análisis Gestionar Cursos. ....	75
Fig 9 Diagrama de clases de análisis Gestionar Perfil. ....	76
Fig 10 Diagrama de clases de análisis Gestionar Información. ....	77
Fig 11 Diagrama de clases de diseño web Autenticar Usuario. ....	78
Fig 12 Diagrama de clases de diseño web Gestionar Cursos. ....	78
Fig 13 Diagrama de clases de diseño web Gestionar Curso Optativo. ....	79
Fig 14 Diagrama de clases de diseño web Gestionar Perfil. ....	80
Fig 15 Diagrama General del funcionamiento de la programación y el paquete Incluye. ....	82
Fig 16 Diagrama de Clases Persistentes. ....	83
Fig 17 Modelo de Datos. ....	84
Fig 18 Diagrama de despliegue. ....	93
Fig 19 Diagrama de componentes general. ....	94
Fig 20 Diagrama de componentes del Paquete Incluye. ....	95
Fig 21 Diagrama de componentes del Paquete Curso. ....	97
Fig 22 Diagrama de componentes del Paquete Curso Optativo. ....	97
Fig 23 Diagrama de componentes del Paquete Perfil. ....	99
Fig 24 Diagrama de componentes Autenticación. ....	101
Fig 25 Representación de la Prueba de Caja negra. ....	102

## INTRODUCCIÓN

Con el desarrollo de la ciencia y la técnica ha aumentado extraordinariamente el conocimiento, apareciendo nuevos conceptos y términos en las ramas del saber. Una de estas ramas es la informática, que cada día, con el desarrollo tecnológico (Internet, comunicaciones móviles, banda ancha, satélites, microondas, etc.), avanza extraordinariamente.

Informática “es el tratamiento racional, automático y adecuado de la información, por medio del computador, para lo cual se diseñan y desarrollan estructuras aplicaciones especiales buscando seguridad e integridad”(Curso de Informática Básica 1997).

Surge aquí un concepto importante que es el de las Tecnologías de la Información y las Comunicaciones (TIC). Actualmente, todos de una forma u otra reconocen la importancia que reviste el empleo de estas TIC; siendo imposible por tanto perfeccionar nuestra labor sin la aplicación de éstas.

Las tecnologías de la información y las comunicaciones se refieren a todos los instrumentos, procesos y soportes que están destinados a optimizar la comunicación humana.

En las empresas, contar con la información oportuna para tomar las mejores decisiones en el momento adecuado, resulta algo fundamental. En esta situación las nuevas tecnologías de la información son muy relevantes, ya que dan la posibilidad de obtener y procesar más datos que los medios manuales.

En Cuba, las TIC se aplican en todas las esferas de la vida, en el sector de la salud, los servicios, la educación, las investigaciones y la gestión económica.

Con el uso de la tecnología en la empresas cubanas, se pretende que ayude en la toma de decisiones, en la seguridad de la información, que satisfaga en gran medida al cliente, que disminuyan los gastos, aumente la productividad, las respuestas sean rápidas, las acciones acertadas, la población tenga un alto desarrollo de su capacidad y conocimientos, etc.

En todas las escuelas del país y el mundo en general, se planifica, organiza y controla la información docente, además de archivarse el recorrido de los estudiantes en las actividades curriculares, pudiéndose obtener reportes con las asignaturas, notas, promedio y otros datos por cada alumno. En algunos centros educacionales del país se imparten cursos optativos de diferentes materias o temas, los cuales son opcionales y contribuyen a la formación profesional e integral del educando. Estos

cursos se comportan como cualquier asignatura, ya que tienen un programa analítico, un profesor, estudiantes que lo reciben, y se planifican las actividades que lo conforman. La UCI, es un centro que se caracteriza por ofertar una gran cantidad de cursos sobre distintas materias (programación, diseño, gestión de calidad, matemática, entre otras), los cuales son impartidos durante todo el año y permiten al estudiante enriquecer sus conocimientos, formar habilidades y capacidades.

Los cursos optativos que se imparten en la Facultad 8 siguen un mecanismo que contemplan varias actividades, como son:

- ✓ La planificación de un horario específico de manera que no coincida con las demás actividades docentes (local, hora, profesor, tipo de curso, frecuencias).
- ✓ Convocatoria de todos los cursos que se ofertarán y planificación de estos.
- ✓ Realizar la matrícula de los estudiantes y profesores interesados (según el caso) teniendo en cuenta que la misma no sobrepase la capacidad del local (laboratorio) y la cantidad de profesores que imparten el curso. En el caso de los estudiantes se hace en Secretaría.
- ✓ Realizar pruebas de suficiencias en caso que se solicite.
- ✓ Impartir los cursos.
- ✓ Acreditar los cursos culminados en la Secretaría docente.
- ✓ Controlar y reportar los cursos en varios aspectos.

Estos procesos se hacen de manera manual, lo que implica esfuerzo humano, gasto de material de oficina, empleo de tiempo, tardanza en la actualización y manipulación de la información referente a los cursos, así como quejas en algunas ocasiones por parte de los implicados o afectados con todo este proceso, sobre todo los estudiantes.

La matrícula en los cursos optativos se hace en la Secretaría, que dada la convocatoria hecha por correo a los estudiantes con fecha límite, ellos pasan por allí para anotarse en un curso, razón que provoca que se anoten varias veces porque no se puede llevar un control si el estudiante vino en un momento dado o en otro, provocando que matrícula no satisfaga lo planificado o previsto inicialmente. Otra forma de ofertar los cursos, es planificarlos para un grupo directamente, de forma tal que la mayor cantidad de estudiante logre matricular en uno. Esto es bueno cuando es de interés que el grupo pase el curso, pero a veces quita la posibilidad de permitir al estudiante optar por un curso en específico. En fin, el proceso de realizar la matrícula para los cursos optativos es inestable, ya que varía en dependencia de las necesidades y el momento en que sean ofertados.

Toda la información referente a los cursos optativos está archivada en papeles en la Secretaría y parte de ella está registrada en Akademos<sup>[1]</sup> (el curso y la nota de cada estudiante), y cada vez que se necesite información de esto, hay que emplear tiempo buscando información y procesándola según las necesidades del interesado, ya que no existe mecanismos ni permisos para hacer un reporte deseado o específico.

Si el responsable de planificar y controlar los cursos optativos, como necesidad suya o de algún directivo de la Facultad desea saber el estado de los cursos, debe pedir esta información en Secretaría o llevar un control propio que coincida con lo que está registrado en actas, para luego procesar los datos y obtener los resultados deseados, lo cual se hace en Microsoft Excel, empleándose un tiempo largo en realizar estas acciones, que trae como consecuencia que el responsable tome parte de su tiempo libre en realizar esta tarea, o no entregue en tiempo lo convenido. También se corre el riesgo de pérdida de la información, ya que esta se guarda en una máquina y puede presentar problemas inesperadamente, aunque en algunos casos, parte de la información se archiva en el correo, pero solo aquella que tenga un tamaño considerable que no llene el buzón del mismo.

Además, en la UCI existe como característica excepcional que cada estudiante debe salir graduado con un perfil (independientemente del título de ingeniero), según la Facultad donde curse la carrera, lo cual es un requerimiento indispensable. Esto tiene como objetivo esencial formar un graduado con conocimientos básicos de la carrera, así como con las capacidades y habilidades sobre un tema determinado que estaría recogido en ese perfil, el cual se compone de asignaturas básicas y optativas que debe recibir y aprobar satisfactoriamente. Si se tiene más de un perfil el estudiante puede escoger con cuál o cuáles graduarse.

En la Facultad # 8, el perfil de Multimedia y Software Educativo (único hasta ahora) fue aprobado en el mes de noviembre del curso escolar 2006-2007, por lo que los directivos de la Facultad y el responsable del Perfil se dieron a la tarea inmediatamente de verificar el estado de los estudiantes en la formación de este perfil, y se tuvo como resultado que ninguno tenía siquiera el 50% de las asignaturas aprobadas. El verificar el estado de cada estudiante tomó un tiempo de una semana aproximadamente (en su mayoría el tiempo era fuera de la jornada laboral), ya que no había una forma rápida de obtener los datos, sino que había que verificar estudiante por estudiante de los 128

---

<sup>[1]</sup> Sistema académico de la Universidad de las Ciencias Informáticas.

que tiene el 5to año actual. Esta información se fue recogiendo en un documento Excel y cada vez que se necesitaba actualizar algo o procesar datos más completos, había que dirigirse a Akademos o verificar la información en las actas archivadas en Secretaría. Si algún directivo de la Facultad o la UCI necesitaban información sobre el estado del perfil, ya sea del 5to año o cualquiera de los restantes años en curso, había que dedicar tiempo en buscar, actualizar y procesar información, y en muchos casos se incumplía la fecha de entrega de la tarea encomendada, ya sea por la escasez de personal o el límite de tiempo para el cumplimiento. Actualmente este es el proceso que se realiza, siendo engorroso y tedioso de realizar durante el curso.

En el caso de que exista más de un perfil – que se pretende que ocurra en el próximo curso escolar – ya el trabajo aumenta, porque el estudiante está en su derecho de escoger cualquiera de estos o estar en más de uno si es su interés, razón que provoca que la información haya que procesarla con más cuidado y lleve más tiempo obtener resultados específicos.

Todas estas limitaciones, problemas y necesidades que dieron origen a esta investigación, afectan a las personas implicadas en el proceso, provocando que el trabajo sea más lento, tedioso y un poco deficiente.

Este trabajo surge por la **necesidad** de gestionar toda la información inherente a los cursos optativos y controlar el estado del perfil de cada estudiante, permitiendo obtener estadísticas sobre datos determinados de dicha información, y con esto tomar decisiones para trazar nuevas estrategias que permitan un mejor desarrollo del trabajo y satisfacción para los implicados. Además, se necesita que los procesos relacionados con la matrícula, la gestión del perfil y los cursos optativos, sean rápidos, cómodos al realizarse, eficientes, seguros y cumplidos en tiempo y con la calidad requerida, contribuyendo con el mejoramiento del trabajo en la Facultad.

**Actualmente**, el trabajo se hace de manera manual y se lleva el control mediante la herramienta Microsoft Excel, que se va actualizando con el objetivo de darle seguimiento al estado de cada estudiante tanto en materia de cursos como de perfil, y de esta forma poder obtener reportes o estadísticas según peticiones o necesidades. Ambas cosas están un poco relacionadas, ya que muchos de los cursos optativos forman parte del perfil que está desarrollando el estudiante, razón por la cual se hace necesario llevar un control de ambas materias. También, la planificación de los cursos y el perfil se hacen dado el estado en que se encuentran los estudiantes en dichas materias, las necesidades que existen, las prioridades, la situación del estudiante docentemente y otros

factores; datos se obtienen generalmente de una búsqueda realizada en Akademos y en el Excel que llevan la secretaria y el responsable de los cursos. Toda la información que se va procesando manualmente se archiva en la máquina donde se trabaja, la cual puede perderse por diferentes factores (rotura de PC o violación de la seguridad), y esto traería consecuencias no deseadas, provocando que se emplee tiempo en volver a hacer el trabajo para posteriores peticiones, gastando más tiempo y personal. Actualmente no existe ninguna vía eficiente y óptima que permita al proceso desarrollarse exitosamente a lo largo del curso, y facilitarle el trabajo al personal involucrado.

Luego de analizar la situación problemática, surge el siguiente **problema**: ¿Cómo facilitar de forma eficiente, segura y rápida el proceso de gestión de la información relacionada con el perfil y los cursos optativos en la Facultad # 8 de la Universidad de las Ciencias Informáticas? Este problema se enmarca en el **objeto de estudio**: proceso de gestión de información académica en la UCI. Para resolver el problema se propone el siguiente **objetivo**: desarrollar un subsistema informático que gestione el proceso relacionado con el perfil y los cursos optativos en la Facultad # 8 de la Universidad de las Ciencias Informáticas.

El objetivo delimita el siguiente **campo de acción**: proceso de gestión de la información sobre el perfil y los cursos optativos en la Facultad # 8.

Como **objetivos específicos** se plantean los siguientes:

- ✓ Procesar toda la información inherente a la gestión de los cursos optativos y el perfil.
- ✓ Realizar el análisis, diseño, implementación y prueba de un subsistema que gestione el proceso relacionado con el perfil y los cursos optativos.
- ✓ Crear un documento que recoja todo el proceso investigativo del desarrollo del sistema informático.

Para cumplir con los objetivos de esta investigación se desarrollaron las siguientes **tareas**:

1. Realizar entrevistas para saber cómo se quiere que funcione el sistema a implementar.
2. Estudiar y analizar toda la información relacionada con el proceso de gestión del perfil y los cursos optativos.
3. Estudiar las metodologías más utilizadas en el mundo y escoger la más adecuada para el desarrollo del producto, así como el lenguaje de modelado y la herramienta informática, para

en su conjunto crear los artefactos del sistema durante su ciclo de vida (análisis, diseño, implementación y prueba).

4. Estudiar las tecnologías para la implementación del software y escoger las más adecuadas.
5. Escribir en formato digital y copia dura todo el proceso investigativo del desarrollo del trabajo como resultado de la experiencia.

### ***Aportes prácticos esperados del trabajo***

Con este trabajo se espera que se cumplan las siguientes expectativas:

- ✓ Disminuir un poco el trabajo durante la ejecución de las actividades implicadas en el proceso, y con esto a su vez, disminuir el personal, el gasto de material de oficina y del tiempo empleado.
- ✓ Mejorar el proceso de matrícula de los estudiantes en los cursos optativos de la facultad, permitiendo que sea más rápido, confiable y seguro.
- ✓ Actualizar, controlar y procesar información sobre el perfil y los cursos de forma rápida, y con esto obtener reportes con datos fiables y completos que serían analizados por personas autorizadas según las necesidades o intereses de la Facultad o Universidad en general.
- ✓ Se pretende que el proceso de acreditación de los cursos optativos se haga en menos tiempo, y así quedaría actualizada más rápido parte de la información generada en todo este proceso.
- ✓ Centralizar la información relacionada con el perfil que está pasando cada estudiante.
- ✓ Con lo anteriormente planteado se pretende que aumente la calidad del trabajo en la Facultad #8.

### ***Ideas a defender***

Se quiere que el software permita:

- ✓ Facilitar de manera rápida, organizada, eficiente y segura la actualización, organización, control y procesamiento de la información inherente al perfil y los cursos optativos.
- ✓ Permitir obtener información según las necesidades de los usuarios (teniendo en cuenta los permisos).

- ✓ Permitir que el proceso de matrícula en los cursos sea rápido y organizado.

En el capítulo 1 se aborda lo referente a la fundamentación teórica, donde se describen los conceptos esenciales para comprender el problema, así como las tecnologías, herramientas y lenguaje de programación relacionados con el campo de acción. Se fundamenta la metodología de desarrollo de software utilizada para guiar la investigación.

En el capítulo 2 se explican los procesos del negocio que se desarrollan en la entidad, así como las reglas que se deben cumplir. Se modeló el diagrama de casos de uso del negocio y se realizaron las descripciones de cada caso de uso. Además, se identifican y refinan los requisitos funcionales definidos por el cliente, los cuales están implícitos en los casos de uso del sistema y estos son descritos detalladamente.

En el capítulo 3 se muestran los diagramas de clases del análisis para cada realización de caso de uso del análisis, los diagramas de clases del diseño web para cada realización de caso de uso del diseño y el diagrama de clases persistente con su modelo de datos. Además se describen las clases utilizadas en el diseño.

En el capítulo 4 se muestran los diagramas de componentes que representan las dependencias entre los ficheros que integran el subsistema en su conjunto y por paquetes de componentes, así como un diagrama de despliegue que identifica la estructura física con la que contará el subsistema una vez llegado el proceso de instalación del producto. Además, se describen las pruebas hechas al software, centradas fundamentalmente en las pruebas de caja negra.

## **CAPÍTULO 1 FUNDAMENTACIÓN TEÓRICA**

### **1.1. Introducción**

Hoy en día en el mundo se revoluciona a una velocidad increíble la rama de la informática, razón por la que uno podría dedicar gran parte de su vida al estudio y análisis de esta. El siguiente capítulo es el resultado de una minuciosa y breve investigación acerca de los conceptos esenciales para comprender los procesos de negocio asociados al dominio del problema, de los software existentes asociados al campo de acción, las nuevas tecnologías en este campo, las tendencias existentes en el mundo, lenguajes de implementación y una breve reseña del proceso de desarrollo de software utilizado para darle solución al problema planteado.

### **1.2. Estado del arte**

#### **1.2.1. Conceptos fundamentales**

##### ***Curso optativo***

Un curso optativo es el conjunto de conocimientos organizados de forma estructurada de cualquier materia o tema que reciben los estudiantes de manera optativa u opcional con el objetivo de prepararse profesionalmente, desarrollar habilidades y capacidades. Estos no forman parte del plan de estudio del programa académico, sino que pueden contribuir a la formación del estudiante en su perfil u otro interés del centro. Cada curso tiene un programa analítico que recoge los objetivos, temas, frecuencias, tipo de clases y otros datos. Este puede ser impartido por estudiantes especializados en el tema o profesores. La forma de evaluación es mediante prueba de suficiencia o recibiendo el curso durante un período determinado.

### **Perfil**

Un perfil es el conjunto de conocimientos organizados de forma estructurada de una materia, tema o línea cognitiva específica, que se conforma de asignaturas básicas y cursos optativos que complementan dicho perfil. Este es recibido por los estudiantes de forma obligatoria y aparejado a su carrera, con el objetivo de prepararlos profesionalmente con capacidades y habilidades profundas en un conocimiento específico.

### **Asignatura básica**

Asignatura perteneciente al perfil que debe recibirse de manera obligatoria, las cuales contienen los conocimientos teóricos – prácticos más importantes de la línea del perfil.

#### **1.2.2. Actualidad del trabajo**

Actualmente, en el mundo, existen mecanismos y software que permiten gestionar información docente y controlar mediante estrategias trazadas y planificadas las asignaturas que conforman el plan de estudio de cualquier carrera o cursos optativos que se brindan u ofertan a estudiantes o trabajadores para permitir su capacitación. Aquí se muestran algunas universidades con procesos vinculados al objeto de estudio del trabajo, abordándose brevemente qué se hace respecto a las materias que se imparten en dichos centros docentes.

Los software de gestión de información académica son muy utilizados en cualquier centro de enseñanza a nivel mundial gracias a las comodidades que ellos proporcionan al claustro profesoral, a la familia del estudiantado y a los estudiantes; ejemplos de estos tenemos: el **sistema MentorVox** desarrollado por el GRUPO ESEMTIA, este facilita como aplicación de gestión académica, la introducción y administración de toda la información generada, permitiendo a los distintos perfiles de usuario (profesores, tutores, jefe de estudios, director), rentabilizar al máximo la gestión de dicha información (ESEMTIA 2004). En Paraguay, el **Sistema de Gestión de Servicios (SGS)** es una herramienta utilizada por el **Centro Nacional de Computación (CNC)** para el control, planeamiento y administración de los servicios prestados a usuarios de diversos departamentos, integrando informaciones de las diferentes áreas de la institución: académica, técnica, de investigación y administrativa ((CNC)) 2005). La aplicación de gestión académica de la **Universidad de Santiago de Compostela** es la **XesCampus**, donde los usuarios que desean acceder a la información de

calidad sin la mediación de los técnicos informáticos y tratarla posteriormente de forma flexible, con las herramientas ofimáticas y de análisis que están acostumbrados a manejar (UPM 2006). La **Universidad del Norte en Colombia** apoya toda su gestión académica y administrativa con cuatro grandes Sistemas de Información institucionales, identificada así:

- ✓ **Sistema Académico. AURORA.**
- ✓ **Sistema Administrativo Financiero. SIRIUS.**
- ✓ **Sistema Gestión Humana. SOPHIA.**
- ✓ **Sistema de Gestión Bibliográfica. SIBILA.**

El primero cubre los procesos de admisión, registro académico y cuentas por cobrar de servicios educativos, en los diferentes niveles de la educación: pregrado, postgrado y educación continua. Constituye en la base para la gestión y servicios que prestan a los estudiantes las oficinas responsables de los servicios académicos, facilita a los estudiantes el uso de la Web para los servicios de admisiones, matrícula, consultas de notas e historia académica entre otras y permite a los docentes el manejo de sus cursos y notas utilizando la Web. El segundo sistema **SIRIUS**, soporta las actividades operativas de las áreas administrativas y financieras; permite articular todos los procesos de almacén, compras, activos fijos, presupuesto, contabilidad, tesorería y mantenimiento. El Sistema **SOPHIA**, el cual está siendo desarrollado internamente, se encargará de atender todos los servicios de Gestión Humana de la Universidad como selección de personal, contratación, nómina, Planta de personal entre otros. El sistema de información bibliográfico **SIBILA**, maneja todos los procesos de adquisición, catalogación, circulación y préstamo de los diferentes materiales bibliográficos en la Universidad (NORTE 2004).

En la **Universidad Veracruzana** el **Sistema Integral de Información Universitaria (siiu)** automatiza métodos y procedimientos de operación de la universidad y sistematiza el uso, la explotación y la presentación de la información universitaria en apoyo a sus funciones sustantivas y adjetivas y a la toma de decisiones. Su propósito es hacer eficiente la gestión académica y administrativa para ofrecer servicios de calidad a la comunidad universitaria, propiciando la adopción de modelos educativos más flexibles. Este sistema funciona a través de tres módulos: estudiantes, recursos humanos y recursos financieros (VERACUZ-MEXICO 2004). La aplicación informática utilizada por la **Universidad de Córdoba** para la automatización de la gestión académica es el **Sistema Integral de Gestión Académica (SIGA)**. Esta aplicación, desarrollada por el servicio de

informática de la universidad, se encuentra en continua evolución, ampliando nuevas funcionalidades y adaptándose a las nuevas tecnologías. Las funcionalidades soportadas por el SIGA son muy amplias, abarcan desde la gestión del acceso a la universidad hasta la tramitación de los títulos pasando por la matrícula, actas de examen, expedientes académicos, estadísticas, informes, gestión de becas, convalidaciones (CÓRDOBA 2007).

En la **Facultad de Informática** de la **Universidad Nacional de la Plata** el alumno debe cursar 5 Optativas de ½ punto entre 4to y 5to año, elegidas con una distribución por Área de 3-1-1 (es decir 3 materias de un área y 1 de cada una de las otras dos) entre el Área de Fundamentos, el Área Arquitectura, Sistemas Operativos y Redes y el Área Algoritmos y Lenguajes, la información se gestiona de forma manual (PLATA 2006). En la **Universidad Tecnológica Equinoccial (Ecuador)** los cursos optativos que se ofrecerán a los alumnos serán sobre temas especiales de informática y computación, relacionados con las carreras profesionales que se dictan en las distintas Facultades. Todos los cursos optativos tendrán un costo que será fijado en función del número de horas y del costo unitario establecido para cada año lectivo por las autoridades de la Universidad, para aprobar los cursos optativos dictados, el alumno debe obtener un promedio mayor o igual al 70% de la calificación máxima establecida y acreditar una asistencia no menor al 70% de las clases dictadas. Todo este proceso es de forma manual (EQUINOCCIAL 2007).

En nuestro país en el sistema de universidades del **(MES) Ministerio de Educación Superior** se ha investigado que existen cursos optativos de diferentes materias por cada carrera y que para gestionar este proceso a la hora de almacenar la información de las notas de los estudiantes, no hay ningún software con estos fines. Contando con algunos para la **Gestión Académica GESTACAD**

1.1: Gestión académica universitaria (Universidad de Matanzas) (UMCC 2005).

En la **Universidad de las Ciencias Informáticas** se consta con un sistema de gestión denominado Akademos que por ser para la gestión a nivel general permite el almacenamiento de los datos, por facultad, de los resultados obtenidos en las pruebas realizadas, la matrícula estudiantil, asistencia en clases y causas que describen la situación del ausente, resultados de los cursos optativos, entre otras; este último no se adapta a las necesidades específicas de la facultad ya que no permite de forma rápida saber algunos datos e información determinada de algún estudiante, grupo, facultad o universidad en general, ya que los permisos para acceder a la información de forma inmediata son

restringidos y en muchos casos, nulos. Cada facultad tiene su propia estrategia para controlar todos los procesos relacionados con el perfil y los cursos optativos, ejemplo: en las **Facultades 1, 2, 4, 6, 7** no existe, según los responsables de esta área, ningún software que ellos utilicen con estos fines, sino que lo hacen de forma manual y accediendo en varias ocasiones al sistema Akademos. Por otra parte, en la **Facultad 3**, se tiene como característica, que los Cursos Optativos se inician a partir de 3 formas fundamentales:

- ✓ Se les distribuye por el J' de colectivo a los diferentes grupos y se les pone en el horario como un turno de clase más, esto es para todos los grupos de la facultad.
- ✓ Se hace una solicitud formal desde un proyecto o desde el decanato al J' de Colectivo a partir de un modelo que existe para esto.
- ✓ A partir de la existencia de disponibilidad de que alguien imparta el CO, entonces se abre la matrícula en secretaría y se divulga entre los estudiantes.

Una vez que esté el Programa Analítico del Curso, la persona que lo va a impartir y los estudiantes, este comienza normalmente.

El profesor o AA de CO, al terminar entrega el listado a la secretaria donde hacen el acta, y el profesor pasa después a poner la nota y firmar, en caso de un AA, es el J' de Colectivo el que firma la planilla.

Luego las secretarias se encargan de pasar eso a Akademos. Para sacar estadísticas se tiene una aplicación Desktop sencilla, con una BD en SQL Server y Crystal Report, resolviendo las necesidades que puedan presentarse o información pedida por cualquier directivo o interesado.

En la **Facultad 5** no hay ningún software para realizar este proceso, la información se procesa de forma manual y con el apoyo de la Secretaría Docente.

En la **Facultad 9**, el procedimiento de inscripción y notas es mediante una estrategia para los estudiantes de 5to año, que consiste en darle un cierto % de la capacidad de cada curso a ellos, así como hay cursos destinados a estudiantes de proyecto con alguna que otra capacidad para los demás, hay cursos que son para 4to y 5to año nada más, otros no.

En la **Facultad 10** hay un segundo perfil compuesto por 5 cursos obligatorios y 9 no obligatorios. Todas las convocatorias de los cursos se hacen a través del correo, y se recogen los nombres. Luego se recoge la asistencia, las notas y se entrega la planilla oficial (por el profesor del curso) en la secretaría de la facultad.

Se está haciendo un sistema Web para gestionar los recursos humanos preparados para cada materia. También se están desarrollando niveles de cursos para evaluar o acreditar, por ejemplo: niveles medios y avanzados de Linux, PHP y Perl.

### 1.3. Tecnologías a utilizar

Como **propuesta de solución** se tiene desarrollar una aplicación web que gestione la información relacionada con el perfil y los cursos optativos, generándose los reportes más utilizados y pedidos por la Facultad o la Universidad en general, permitiendo exportarlos a un documento Excel, para una posterior manipulación o impresión si es necesario por parte de los implicados. Para el desarrollo del sistema se investigaron varias tecnologías que se utilizan en el mundo, de las cuales fueron elegidas algunas según sus ventajas y facilidades para dar solución al problema.

#### SERVIDOR HTTP APACHE

El **servidor HTTP Apache** es un servidor de páginas Web que permite acceder a las páginas Web que están alojadas en un ordenador. Es de código abierto y actualmente es el servidor Web que más se utiliza en el mundo, encontrándose por encima de sus competidores, tanto gratuitos como comerciales. Funciona sobre cualquier plataforma. Hoy en día es uno de los mejores servidores en términos de eficiencia, funcionalidad y velocidad.

Este servidor tiene capacidad para servir páginas estáticas como dinámicas a través de otras herramientas soportadas que facilitan la actualización de los contenidos usando bases de datos, ficheros u otras fuentes de información.

La estructura de él es en módulos, es decir, está dividido en muchas porciones de código que hacen referencia a diferentes aspectos o funcionalidades del servidor Web. Esta modularidad es intencionada ya que la configuración de cada módulo se hace mediante la configuración de las directivas que están contenidas dentro del módulo. Los módulos del Apache se pueden clasificar en tres categorías:

- ✓ Módulos Base: Módulo con las funciones básicas del Apache.
- ✓ Módulos Multiproceso: Son los responsables de la unión con los puertos de la máquina, aceptando las peticiones y enviando a los hijos a atender a las peticiones.
- ✓ Módulos Adicionales: Cualquier otro módulo que le añada una funcionalidad al servidor.

Las funcionalidades más elementales se encuentran en el módulo base, siendo necesario un módulo multiproceso para manejar las peticiones. Se han diseñado varios módulos multiprocesos para cada uno de los sistemas operativos sobre los que se ejecuta el Apache, optimizando el rendimiento y rapidez del código.

El resto de funcionalidades del servidor se consigue por medio de módulos adicionales que se pueden cargar. Para añadir un conjunto de utilidades al servidor, simplemente hay que añadirle un módulo, de forma que no es necesario volver a instalar el software.

**Internet Information Services (o Server)**, IIS, es una serie de servicios para los ordenadores que funcionan con Windows. Originalmente era parte del *Option Pack* para Windows NT. Luego fue integrado en otros sistemas operativos de Microsoft destinados a ofrecer servicios, como Windows 2000 o Windows Server 2003. Windows XP Profesional incluye una versión limitada de IIS. Los servicios que ofrece son: FTP, SMTP, NNTP y HTTP/HTTPS.

Este servicio convierte a un computador en un servidor de internet o Intranet, es decir, que en las computadoras que tienen este servicio instalado se pueden publicar páginas web tanto local como remotamente (servidor web).

El servidor web se basa en varios módulos que le dan capacidad para procesar distintos tipos de páginas, por ejemplo Microsoft incluye los de Active Server Pages (ASP) y ASP.NET. También pueden ser incluidos los de otros fabricantes, como PHP o Perl.

**THHTTPD** es simple, pequeño, portátil, rápido, y seguro, ya que utiliza los requerimientos mínimos de un servidor HTTP.

La creación de este software colabora para la optimización de las aplicaciones en los servidores WEB, obviamente es tener un hardware potente con un software optimizado y ligero, porque siempre se intenta buscar esa combinación.

## PHP

PHP es un lenguaje de programación usado generalmente para la creación de contenido para sitios Web. Las iniciales PHP significan "PHP Hypertext Pre-processor" y se trata de un lenguaje de programación que es usado para la creación de aplicaciones para servidores, o creación de contenido dinámico para sitios Web.

Ventajas que trae trabajar con php:

- ✓ Es un lenguaje multiplataforma.
- ✓ Capacidad de conexión con la mayoría de los manejadores de base de datos que se utilizan en la actualidad, destaca su conectividad con MySQL y Postgre.
- ✓ Posee una amplia documentación en Internet, incluyendo una gran variedad de ejemplos y de ayudas.
- ✓ Es libre, por lo que se presenta como una alternativa de fácil acceso para todos.
- ✓ Permite las técnicas de Programación Orientada a Objetos.
- ✓ Permite crear los formularios para la Web.
- ✓ No requiere definición de tipos de variables ni manejo detallado del bajo nivel, facilitando un poco la programación.

## PostgreSQL

PostgreSQL: Es un Sistema gestor de bases de datos. Liberado bajo la licencia BSD, que posee una serie de características positivas respecto a otros.

1. Gran escalabilidad. Es ajustable al número de procesadores y a la cantidad de memoria que posee el sistema de la forma más óptima, por este motivo es capaz de soportar una mayor cantidad de peticiones simultáneas. Teniendo en cuenta esto, es vital en la universidad, ya que no se requiere de una gran computadora para trabajar con él.
2. Tiene la capacidad de almacenar procedimientos en la propia base de datos.
3. Multiusuario, multiprogramado, con arquitectura cliente-servidor y control de privilegios de acceso.
4. Los tipos internos han sido mejorados, incluyendo nuevos tipos de fecha/hora de rango amplio y soporte para tipos geométricos adicionales.

5. La velocidad del código del motor de datos ha sido incrementada aproximadamente en un 20% - 40%, y su tiempo de arranque ha bajado el 80% desde que la versión 6.0 fue lanzada.

### **Smarty**

Es un motor de plantillas para PHP, cuyo objetivo es separar el contenido de la presentación en una página web, se encuentra bajo la licencia LGPL por lo que puede ser usado libremente. Lo interesante es que en la programación con PHP existe la tendencia a combinar las labores de diseñador gráfico y de programador en una persona y dentro del mismo código, teniéndose que escarbar entre los scripts para modificar la presentación del contenido, lo que trae consigo grandes dificultades a la hora de cambiar alguna parte del diseño de la página, es por eso que como se hace en grandes proyectos donde los roles son cubiertos por personas distintas, surge Smarty como tecnología a solucionar este problema (WIKIPEDIA 2007b). Para que se entienda mejor, si el diseñador de la plantilla quiere rediseñarla en su totalidad, estos cambios no afectaran la aplicación lógica. Por lo tanto, el programador puede hacer cambios en la aplicación lógica sin que sea necesario reestructurar la plantilla, y el diseñador de la plantilla puede hacer cambios sin que haya dificultad con la aplicación lógica. Por otra parte Smarty no intenta separar completamente la lógica de la plantilla. No hay problema entre la lógica y su plantilla bajo la condición que esta lógica sea estrictamente para presentación. Un consejo: mantener la aplicación lógica fuera de la plantilla, y la presentación fuera de la aplicación lógica. Esto tiene como finalidad tener un objeto más manipulable y escalable para un futuro próximo. Un único aspecto acerca de Smarty es la compilación de la plantilla. De esta manera Smarty lee la plantilla y crea los scripts de PHP. Una vez creados, son ejecutados sobre él (OHRT *et al.* 2004).

Algunas características de *Smarty*:

- ✓ Es extremadamente rápido.
- ✓ Es eficiente.
- ✓ No analiza gramaticalmente desde arriba el template, solo compila una vez.
- ✓ Él está atento para solo recompilar los archivos de plantilla que fueron cambiados.
- ✓ Usted puede crear funciones habituales y modificadoras de variables customizadas, de modo que el lenguaje de la platilla es altamente extensible.
- ✓ Sintaxis de etiquetas delimitadoras para configuración de la plantilla.

- ✓ Los constructores if/elseif/else/endif son pasados por el interpretador de PHP.
- ✓ Es posible incrustar directamente código PHP en los archivos de plantilla, aunque esto puede no ser necesario (no recomendado) dado que la herramienta se puede ajustar.

### **Librería ADOdb**

Esta clase permite la conexión de las aplicaciones web a los sistemas gestores de base de datos debido a un conjunto de funcionalidades que se le han ido implementando, es fácil para programadores Windows debido a que muchas de las convenciones son similares al ADO de Microsoft. Actualmente incluye plug-in para MySQL, Oracle, Microsoft SQL Server, Sybase, Sybase SQL Anywhere, Informix, PostgreSQL, FrontBase, SQLite, Interbase (versiones de Firebird y Borland), Foxpro, Access, ADO, DB2, SAP DB u ODBC. Además se puede conectar a Progres y CacheLite via ODBC. Se puede almacenar la información de sesión usando ADOdb para tener una verdadera portabilidad y escalabilidad. En ADOdb a diferencia de otras clases PHP de base de datos que se enfocan únicamente en el enunciado SELECT. ADOdb soporta código para manejar INSERT y UPDATE que son rápidamente adaptables a múltiples bases de datos. También hay métodos para manejo de fechas, concatenación de cadenas y encomillado de cadenas para diferentes bases de datos. Tiene un sistema de metatipos (metatype) para poder determinar cuáles tipos como CHAR, TEXT o STRING son equivalentes en diferentes bases de datos. Es fácil de portar debido a que todo el código que depende de la base de datos está en funciones. No tiene que portar la lógica principal de las clases.

Creación de tablas e índices portables con las clases de diccionario de datos datadict. Monitor de rendimiento de base de datos y ajuste de enunciados SQL con la clase de performance monitoring. Sesiones en base de datos con la clase session management. Maneja notificación por sesión vencida. Mapeo a Objetos Relacionales usando la clase ADOdb\_Active\_Record (LIM 2000).

### **Java Script**

Es un lenguaje interpretado, es decir, que no requiere compilación, utilizado principalmente en páginas web, con una sintaxis semejante a la del lenguaje Java y el lenguaje C. Al contrario que Java, JavaScript no es un lenguaje orientado a objetos propiamente dicho, ya que no dispone de Herencia, es más bien un lenguaje basado en prototipos, ya que las nuevas clases se generan clonando las clases base (prototipos) y extendiendo su funcionalidad. Todos los navegadores

interpretan el código JavaScript integrado dentro de las páginas web (WIKIPEDIA 2007a). El JavaScript permite crear aplicaciones específicamente orientadas a su funcionamiento en Internet. Usando JavaScript, se pueden crear páginas HTML dinámicas que procesen la entrada del usuario y que sean capaces de gestionar datos persistentes usando objetos especiales, archivos y bases de datos relacionales. Con JavaScript se pueden construir aplicaciones que varían desde la gestión de la información corporativa interna y su publicación en Intranets hasta la gestión masiva de transacciones de comercio electrónico. JavaScript puede utilizar una tecnología propietaria de Netscape, denominada LiveConnect; con el propósito de que las aplicaciones JavaScript puedan tener acceso a aplicaciones basadas en objetos distribuidos CORBA y Java. En cualquier caso, es importante señalar que, pese a la similitud de nombres, JavaScript no es Java. Se utiliza mayormente para lograr una validación que no recargue las funciones del servidor web, ya que permite que se validen los datos del formulario a través del código de la página antes de que esta los enviara a ser procesados (FACTORÍA DE INTERNET S.L. 2007).

Las tecnologías anteriores fueron utilizadas en la construcción de la solución propuesta, por las ventajas que reportan para lo que se quiere lograr:

- ✓ Para la programación el lenguaje escogido es el PHP por ser multiplataforma, libre y permitir las técnicas de Programación Orientada a Objetos (POO). Además, PHP posee una amplia documentación en Internet incluyendo una gran variedad de ejemplos y de ayudas, además de que se destaca por su conectividad con MySQL y Postgre.
- ✓ Como gestor de base de datos fue escogido PosgreSql, ya que tiene la capacidad de almacenar procedimientos en la propia base de datos, es multiusuario, multiprogramado, con arquitectura cliente-servidor y permite el control de privilegios de acceso, además de que es ajustable al número de procesadores y a la cantidad de memoria que posee el sistema de la forma más óptima, por este motivo es capaz de soportar una mayor cantidad de peticiones simultáneas. Teniendo en cuenta esto, este gestor sirve en la UCI porque no se requiere de una computadora potente para trabajar con él.
- ✓ Como Servidor http se eligió el Apache, ya que es el servidor Web que más se utiliza en el mundo, encontrándose por encima de sus competidores, tanto gratuitos como comerciales. Además, funciona sobre cualquier plataforma, y hoy en día es uno de los mejores servidores en términos de eficiencia, funcionalidad y velocidad, teniendo la capacidad para servir

páginas estáticas como dinámicas a través de otras herramientas soportadas que facilitan la actualización de los contenidos usando bases de datos, ficheros u otras fuentes de información.

#### **1.4. Arquitectura utilizada**

##### ***Arquitectura***

Una arquitectura es el conjunto de decisiones significativas sobre la organización del sistema software, la selección de los elementos estructurales y sus interfaces, con los que se compone el sistema, junto con su comportamiento tal como se especifica en las colaboraciones entre esos elementos, la composición de esos elementos estructurales y de comportamiento en subsistemas progresivamente más amplios, y el estilo de arquitectura que guía esta organización estos elementos y sus interfaces, sus colaboraciones, y su composición. Ej: los Patrones de diseño relacionados con el diseño de los objetos y frameworks de pequeña y mediana escala, que son aplicables al diseño de una solución para conectar los elementos de gran escala que se definen mediante los patrones de arquitectura, y durante el trabajo de diseño detallado para cualquier aspecto del diseño local. También se conocen como patrones de micro-arquitectura. El patrón Fachada, que se puede utilizar para proporcionar la interfaz de una capa a la siguiente (LARMAN 1999).

##### ***Arquitectura en Capas***

La **programación por capas** es un estilo de programación en la que el objetivo primordial es la separación de la lógica de negocios de la lógica de diseño. La ventaja principal de este estilo, es que el desarrollo se puede llevar a cabo en varios niveles y en caso de algún cambio sólo se ataca al nivel requerido sin tener que revisar entre código mezclado. Ejemplos de este método de programación se explican a continuación.

##### **1. Arquitectura de 2 capas**

En una aplicación tradicional de 2 capas, la carga de procesamiento es facilitada a la PC cliente mientras el servidor actúa simplemente como controlador del tráfico entre la aplicación y los datos.

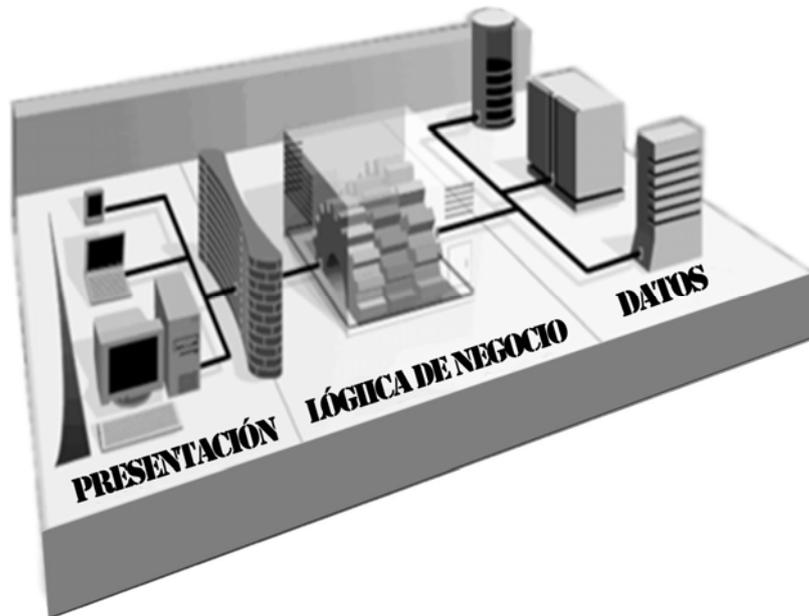
Como resultado amigo tenemos que el rendimiento de la aplicación tiende a aumentar. Cuando la aplicación completa es procesada en una PC, la aplicación es forzada a realizar múltiples peticiones de datos antes incluso de presentar algo al usuario, estas múltiples peticiones de base de datos pueden sobrecargar la red y sobrecargar tu motor de BD.

## **2. Arquitectura de 3 capas**

Para enfrentarse a estos temas, la comunidad de software desarrolló la noción de una arquitectura de 3 capas. Es decir una aplicación se divide en 3 capas lógicas distintas, cada uno de ellas con un grupo de interfaces perfectamente definido. La primera capa se le denomina *capa de presentación* y consiste en la Interfaz Gráfica de Usuario (GUI). La capa intermedia se le denomina *capa de Lógica de Negocio*, consisten en las reglas del negocio sobre el sistema en desarrollo, y la tercera capa, la *capa de Datos*, contiene la Base de Datos para el sistema en desarrollo.

Entonces se dice que la Lógica del negocio sirve para recoger los datos de la capa de datos, formatearlos para luego pasárselos a la capa de presentación, esto permite tener múltiples interfaces sin tener que modificar la lógica del negocio (PERU 2006).

Otro término utilizado es el nivel, por lo que "capas" y "niveles" no significan lo mismo ni son similares. El término "capa" hace referencia a la forma como una solución es segmentada desde el punto de vista lógico: (Presentación/ Lógica de Negocio/ Datos). En cambio, el término "nivel", corresponde a la forma en que las capas lógicas se encuentran distribuidas de forma física. Por ejemplo: Una solución de tres capas (presentación, lógica, datos) que residen en un solo ordenador (Presentación+lógica+datos), se dice, que la arquitectura de la solución es de tres capas y un nivel. Una solución de tres capas (presentación, lógica, datos) que residen en dos ordenadores (presentación+lógica, lógica+datos), se dice que la arquitectura de la solución es de tres capas y dos niveles. Una solución de tres capas (presentación, lógica, datos) que residen en tres ordenadores (presentación, lógica, datos), la arquitectura que la define es: solución de tres capas y tres niveles.



**Fig 1** Arquitectura de 3 capas.

La propuesta de solución está basada en la arquitectura de tres capas y tres niveles, porque esto permitirá una mayor fluidez del trabajo, ya que los errores que puedan surgir en una capa son transparentes a otras capas, solo es necesario corregir en ese lugar con más certeza. Este patrón permite además implementar interfaces de usuario para otro tipo de dispositivos utilizando la lógica de negocio y la capa de datos existentes. Además, tomando en cuenta que se cuenta con dos computadoras: una para el servidor de base de datos y otra para la aplicación, esto le daría, junto con las computadoras en las cuales se navegará por la red en la aplicación, los niveles que se plantean.

## 1.5. Herramientas utilizadas en la propuesta de solución.

### **NuSphere PhpED**

Es un ambiente de desarrollo Integrado (IDE, Integrated Development Environment) para PHP, un idioma común para scripts HTML del lado del servidor. PhpED una flexible y fácil plataforma para sitios web en vías de desarrollo que usan PHP, Perl, Pitón, JavaScript, C++, CSS (Cascading Style Sheet), y HTML. PhpED también mantiene la funcionalidad de poner a punto el código, publicando los proyectos en los servidores remotos a través de FTP y WebDAV, trabajando con los servidores SOAP. PhpED es conveniente para los diseñadores d proyectos muy grandes (CORPORATION 2003).

### **Dreamweaver 8**

Es uno de los editores de desarrollo Web más utilizado a nivel profesional para la creación de sitios Web. Su amplio abanico de herramientas permite crear desde la más simple página Web personal hasta el sitio Web más completo y complejo para una gran empresa y utilizar casi todos los recursos de la Web.

Este editor de HTML profesional para el diseño, codificación y desarrollo de páginas, sitios y aplicaciones Web; permite la edición visual, o sea, crear páginas rápidamente sin escribir una línea de código, así como también la codificación manual (MÉNDEZ and TORRES junio 2005).

Dreamweaver ayuda a construir aplicaciones Web dinámicas apoyadas en bases de datos.

Es completamente personalizable. Se pueden crear objetos y comandos propios, modificar los accesos directos de teclado, e incluso escribir código JavaScript para extender las capacidades del Dreamweaver con nuevos comportamientos. Soporta varias tecnologías del servidor para la construcción de aplicaciones Web, tales como: Macromedia ColdFusion, Microsoft ASP, Microsoft ASP.NET, Sun JavaServer Pages (JSP) y PHP.

### **Rational Rose**

Existen herramientas CASE de trabajo visuales como el Analise, el Designe, el Rational Rose, que permiten realizar el modelado del desarrollo de los proyectos, en la actualidad la mejor y más utilizada en el mercado mundial es Rational Rose. Cubre todo el ciclo de vida de un proyecto: concepción y formalización del modelo, construcción de los componentes, transición a los usuarios y certificación de las distintas fases y entregables. Es la herramienta CASE que comercializan los desarrolladores de UML y que soporta de forma completa la especificación del UML. Rose es una herramienta con plataforma independiente que ayuda a la comunicación entre los miembros de equipo, a monitorear el tiempo de desarrollo y a entender el entorno de los sistemas. Una de las grandes ventajas de Rose es que utiliza la notación estándar en la arquitectura de software(UML), la cual permite a los arquitectos de software y desarrolladores visualizar el sistema completo utilizando un lenguaje común, además los diseñadores pueden modelar sus componentes e interfaces en forma individual y luego unirlos con otros componentes del proyecto (PÉREZ and SÁNCHEZ junio 2005).

### **Firework 8**

Esta herramienta se utiliza para crear, editar y animar gráficos Web, añadir interactividad avanzada y optimizar imágenes en entornos profesionales. En Fireworks es posible crear y modificar imágenes vectoriales y de mapa de bits en una sola aplicación. Fireworks se integra con otros productos de Macromedia, como Dreamweaver, Flash, FreeHand y Director, y con otros editores HTML y aplicaciones gráficas de uso frecuente para ofrecer una solución Web global. Los gráficos de Fireworks pueden exportarse fácilmente con código HTML y JavaScript adaptado al editor de HTML que se utilice. Sirve para optimizar de forma precisa imágenes para la Web en un entorno intuitivo y personalizable. Además admite formatos ActionScript y CSS (Cascading Style Sheet), lo que permite a la aplicación trabajar de una manera incluso más eficaz con Dreamweaver y Flash (MACROMEDIA 2002).

## 1.6. Metodologías de desarrollo de software

En la actualidad, son muchos los procesos y metodologías de desarrollo de software que existen, los cuales, si son bien definidos y correctamente utilizados, traen resultados satisfactorios como: alta calidad en el producto que se obtiene, con un bajo costo y eficaz, óptima planificación que dan cumplimiento en la fecha establecida, etc. Un número creciente de herramientas automatizadas han surgido para ayudar a definir y aplicar un proceso de desarrollo de software efectivo. Hoy en día la economía global depende más de sistemas automatizados que en épocas pasadas; esto ha llevado a los equipos de desarrollo a enfrentarse con una nueva década de procesos y estándares de calidad.

La Ingeniería de Requerimientos cumple un papel primordial en el proceso de producción de software, ya que enfoca un área fundamental: la definición de lo que se desea producir. Su principal tarea consiste en la generación de especificaciones correctas que describan con claridad, sin ambigüedades, en forma consistente y compacta, el comportamiento del sistema; de esta manera, se pretende minimizar los problemas relacionados al desarrollo de sistemas.

Los procesos de desarrollo, se utilizan para trabajar eficientemente, evitar catástrofes que llevan a que un gran porcentaje de proyectos terminen sin éxito. El objetivo de un proceso de desarrollo es subir la calidad del software (en todas las fases por las que pasa) a través de una mayor transparencia y control sobre el proceso. Da igual si es algo casero o para un cliente, hay que producir lo esperado en el tiempo y coste esperados. Es labor del proceso de desarrollo hacer que esas medidas para aumentar la calidad sean reproducibles en cada desarrollo. La implantación de un proceso de desarrollo es una labor más a medio-largo plazo que una labor de resultados inmediatos. Cuesta tiempo que los trabajadores se adapten al proceso, pero una vez superado la inversión se recupera con creces. Es por ello que no tiene sentido ajustarse a un proceso al pie de la letra, sino que hay que adaptarlo a las necesidades y características de cada empresa, equipo de trabajo o casi a cada proyecto.

Aquí se describen las principales características de tres de las más famosas y conocidas metodologías de desarrollo de software: Proceso Unificado de Racional (Rational Unified Process, RUP), Programación Extrema (Extreme Programming, XP) y Desarrollo Guiado por la Funcionalidad (Feature Driven Development, FDD).

## RUP

La metodología **RUP**, llamada así por sus siglas en inglés Rational Unified Process, divide en 4 fases el desarrollo del software:

- ✓ **Inicio:** El objetivo en esta etapa es determinar la visión del proyecto.
- ✓ **Elaboración:** En esta etapa se determina la arquitectura óptima.
- ✓ **Construcción:** En esta etapa se llega a obtener la capacidad operacional inicial.
- ✓ **Transmisión:** El objetivo es llegar a obtener el release del proyecto.

Cada una de estas etapas es desarrollada mediante el ciclo de iteraciones, la cual consiste en reproducir el ciclo de vida en cascada a menor escala. Los Objetivos de una iteración se establecen en función de la evaluación de las iteraciones precedentes.

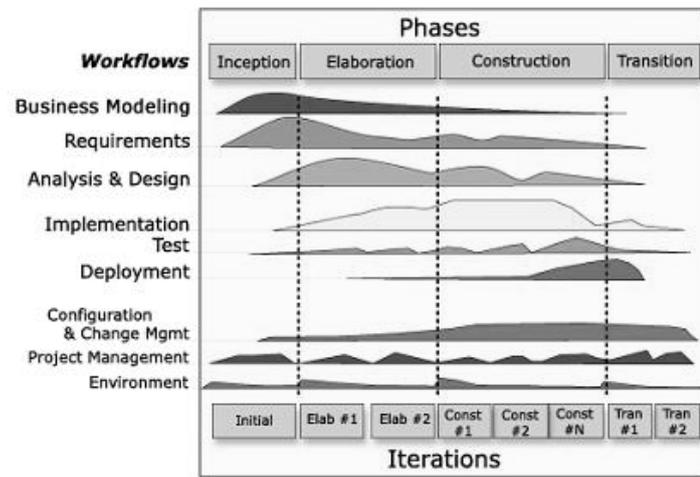
Vale mencionar que el ciclo de vida que se desarrolla por cada iteración, es llevada bajo dos disciplinas:

### **Disciplina de Desarrollo**

- ✓ Ingeniería de Negocios: Entendiendo las necesidades del negocio.
- ✓ Requerimientos: Traslado de las necesidades del negocio a un sistema automatizado.
- ✓ Análisis y Diseño: Traslado de los requerimientos dentro de la arquitectura de software.
- ✓ Implementación: Creando software que se ajuste a la arquitectura y que tenga el comportamiento deseado.
- ✓ Pruebas: Asegurándose que el comportamiento requerido es el correcto y que todo lo solicitado está presente.

### **Disciplina de Soporte**

- ✓ Configuración y administración del cambio: Guardando todas las versiones del proyecto.
- ✓ Administrando el proyecto: Administrando horarios y recursos.
- ✓ Ambiente: Administrando el ambiente de desarrollo.
- ✓ Distribución: Hacer todo lo necesario para la salida del proyecto



**Fig 2** Fases e Iteraciones de RUP.

Es recomendable que a cada una de estas iteraciones se les clasifique y ordene según su prioridad, y que cada una se convierta luego en un entregable al cliente. Esto trae como beneficio la retroalimentación que se tendría en cada entregable o en cada iteración.

**Los elementos del RUP son:**

- ✓ Actividades: Son los procesos que se llegan a determinar en cada iteración.
- ✓ Trabajadores: Son las personas o entes involucrados en cada proceso.
- ✓ Artefactos: Un artefacto puede ser un documento, un modelo, o un elemento de modelo.

Una particularidad de esta metodología es que, en cada ciclo de iteración, se hace exigente el uso de artefactos, siendo por este motivo, una de las metodologías más importantes para alcanzar un grado de certificación en el desarrollo del software.

**XP**

XP, es una metodología ágil, intenta reducir la complejidad del software por medio de un trabajo orientado al objeto, basado en las relaciones interpersonales y la velocidad de reacción. Intenta minimizar el riesgo de fallo del proceso por medio de la disposición permanente de un representante competente del cliente a disposición del equipo de desarrollo. Este representante debe estar en

condiciones de contestar rápida y correctamente a cualquier pregunta del equipo de desarrollo de forma que no se retrase la toma de decisiones. En base a su opinión se definen las siguientes iteraciones del proyecto y si el cliente no está contento se adaptará el plan de releases e iteraciones hasta que el cliente dé su aprobación y el software esté a su gusto. UserStories y casos de pruebas son la base sobre la que se asienta el trabajo del desarrollador. Se sigue un diseño evolutivo con la siguiente premisa: conseguir la funcionalidad deseada de la forma más sencilla posible. Este diseño evolutivo hace que no se le dé apenas importancia al análisis como fase independiente, puesto que se trabaja exclusivamente en función de las necesidades del momento (MOLPECERE 2002).

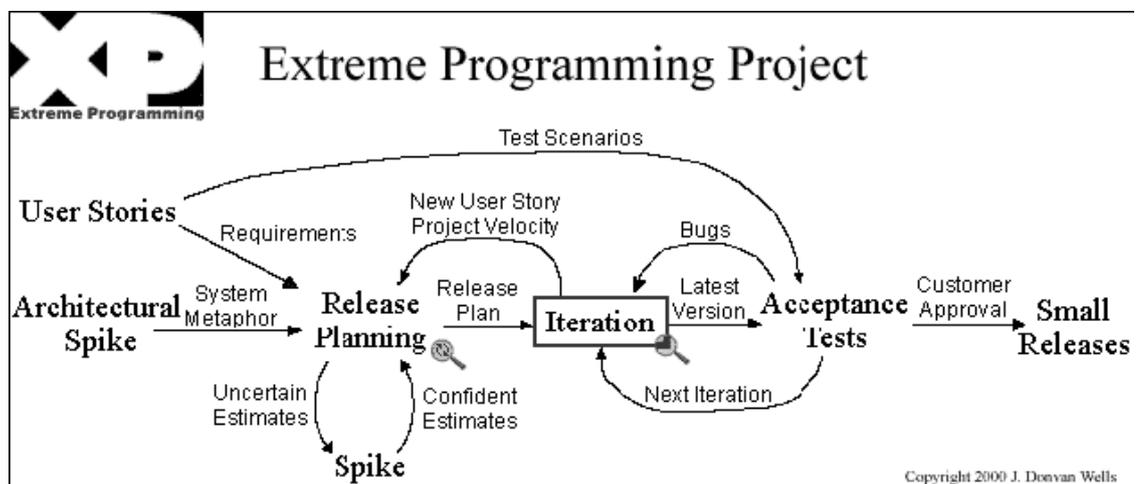


Fig 3 Metodología XP.

Es una de las metodologías de desarrollo de software más exitosas en la actualidad utilizadas para proyectos de corto plazo, corto equipo y cuyo plazo de entrega era ayer. La metodología consiste en una programación rápida o extrema, cuya particularidad es tener como parte del equipo, al usuario final, pues es uno de los requisitos para llegar al éxito del proyecto.

## FDD

FDD se puede considerar a medio camino entre RUP y XP, aunque al seguir siendo un proceso ligero, se asemeja más a XP. Está pensado para proyectos con tiempo de desarrollo relativamente cortos (menos de un año). Se basa en un proceso iterativo con iteraciones cortas (2 semanas). Las iteraciones se deciden en base a funcionalidades, que son pequeñas partes del software con significado para el cliente. Presenta las 5 fases:

1. Desarrollo de un modelo general.
2. Construcción de una lista de funcionalidades.
3. Plan de releases (versiones) en base a las funcionalidades a implementar.
4. Diseñar en la base a las funcionalidades.
5. Implementar en base a las funcionalidades.

Las primeras tres fases ocupan gran parte del tiempo en las primeras iteraciones, siendo las dos últimas las que absorben la mayor parte del tiempo según va avanzando el proyecto, limitándose las primeras a un proceso de refinamiento. Las funcionalidades a implementar en un release (versión) se dividen entre los distintos subgrupos del equipo, y se procede a implementarlas (MOLPECERE 2002).

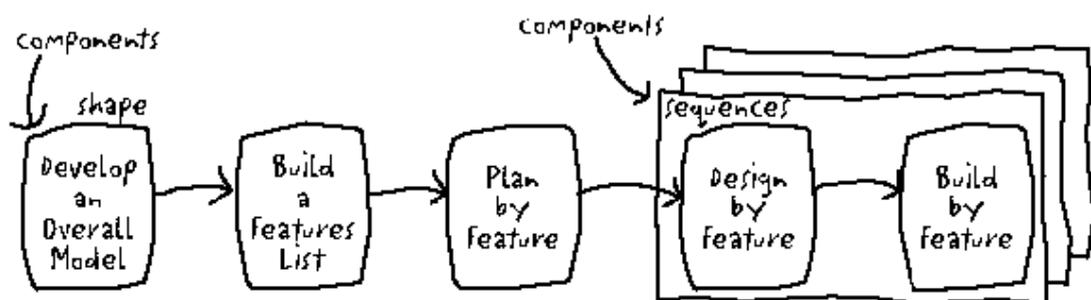


Fig 4 Metodología FDD.

### **Metodología utilizada para la propuesta de solución**

Después de realizado un análisis e investigación de estas tres metodologías, se elige utilizar RUP para el desarrollo del software, ya que es la más completa y abarcadora, pues como señalan algunos autores, las otras metodologías son casos particulares de esta. Además, XP y FDD presentan algunas debilidades, lo que representa riesgos considerables, como dificultades a la hora de una buena obtención de requisitos para el sistema. RUP es adaptable, según el tipo de proyecto, así serán las características del mismo, haciéndose más énfasis en aquellos flujos de trabajo durante la vida del software, que reporten más importancia y sean indispensables. RUP permite trazarse planes de riesgos y pruebas durante el ciclo de vida. Contiene artefactos que son diseñados durante las diferentes fases, los cuales describen detalladamente las características del software desde que se realiza el análisis del problema hasta la entrega final del producto.

### **UML**

El Proceso Unificado RUP (Racional Unified Process) utiliza el Lenguaje de Modelado UML (Unified Modeling Language) para preparar todos los esquemas de un sistema software. UML es un parte esencial del Proceso Unificado.

UML es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad. Es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema de software. UML ofrece un estándar para describir un "plano" del sistema (modelo), incluyendo aspectos conceptuales tales como procesos de negocios y funciones del sistema, y aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y componentes de software reutilizables. Es una notación estándar para el modelado de sistemas software, resultado de una propuesta de estandarización promovida por el consorcio OMG (Object Management Group), del cual forman parte las empresas más importantes que se dedican al desarrollo de software, en 1996. UML es una notación, es decir, una serie de reglas y recomendaciones para representar modelos. UML recomienda 9 diagramas que son modelados durante las diferentes fases de desarrollo del software, es decir, representan las siguientes vistas del sistema:

- ✓ Diagrama de Casos de Uso: modela la funcionalidad del sistema agrupándola en descripciones de acciones ejecutadas por un sistema para obtener un resultado.

- ✓ Diagrama de Clases: muestra las clases que componen el sistema y cómo se relacionan entre sí.
- ✓ Diagrama de Objetos: muestra una serie de objetos y sus relaciones.
- ✓ Diagrama de Secuencia: enfatiza la interacción entre los objetos y los mensajes que intercambian entre sí junto con el orden temporal de los mismos.
- ✓ Diagrama de Colaboración: igualmente, muestra la interacción entre los objetos resaltando la organización estructural de los objetos en lugar del orden de los mensajes intercambiados.
- ✓ Diagrama de Estados: modela el comportamiento de acuerdo con eventos.
- ✓ Diagrama de Actividades: simplifica el Diagrama de Estados modelando el comportamiento mediante flujos de actividades.
- ✓ Diagrama de Componentes: muestra la organización y las dependencias entre un conjunto de componentes.
- ✓ Diagrama de Despliegue: muestra los dispositivos que se encuentran en un sistema y su distribución en el mismo.

## **1.7. Conclusiones**

En este capítulo quedaron reflejados todos los conceptos relacionados con el problema, ampliando así los conocimientos para comprender mejor los procesos de negocio que ocurren en la entidad. Se hizo un análisis del estado del arte, las tecnologías y herramientas informáticas, procesos de desarrollo de software, lenguajes de programación y tendencias que fundamentan la solución propuesta.

## **CAPÍTULO 2 CARACTERÍSTICAS DEL SISTEMA**

### **2.1. Introducción**

En el siguiente capítulo se describen las reglas del negocio asociadas al dominio del problema, así como los flujos de trabajo, ofreciendo una visión general del funcionamiento de la entidad. Se hace referencia a los requisitos funcionales y no funcionales que debe cumplir el sistema que se propone, lo que permite hacer una concepción general del mismo. Además, nos permite identificar mediante un Diagrama de Casos de Uso, las relaciones de los actores que interactúan con el sistema, reflejándose las acciones que se ejecutan o llevan a cabo.

### **2.2. Reglas del negocio.**

En la Universidad de las Ciencias Informáticas (UCI) se tiene como característica que cada estudiante tiene que graduarse de ingeniero informático y con un perfil en dependencia de la Facultad donde desarrolla la carrera dentro de la universidad. Para poder aprobar el perfil, el estudiante debe recibir varios cursos básicos y optativos que le tributan a este, y otros que le permiten aumentar su conocimiento y desarrollar habilidades y capacidades, y de forma general, todos tributan al promedio final.

En relación con esto, el negocio comienza por la necesidad de que los estudiantes tengan acreditados cursos optativos y aprobado al menos un perfil. Para organizar los cursos a ofertar, el responsable de estos (actualmente es la Jefa de Colectivo de la asignatura de Práctica Profesional) realiza una reunión preliminar con todos los alumnos ayudantes (AA) y profesores que van a impartir los cursos, con el objetivo de definir, planificar y organizar los cursos optativos que se ofertarán, así como trazar una estrategia para controlar y gestionar los riesgos que puedan surgir durante el período. Además, se propone un horario de conveniencia para cada AA o profesor que impartirá el curso (siempre que se pueda), permitiendo que se cumpla la tarea con la calidad requerida y sin afectaciones. De estos profesores se recogen los siguientes datos: nombre, apellidos, facultad, categoría (profesor, especialista, AA) y curso a impartir.

Una vez planificado todo lo referente a los cursos, se lanza una convocatoria para realizar los exámenes de suficiencia, con el objetivo de que el estudiante elija de forma abierta los cursos que

desea examinar (en un plazo determinado), permitiendo practicar sus conocimientos y poder acreditarlos por esta vía. Una vez vencido el plazo, se les comunica a los que impartirán los cursos solicitados, que elaboren un examen y se planifica un horario durante 1 o dos días (según la cantidad de estudiantes) para realizarlo, informándose posteriormente a todos los implicados por vía correo. En el mismo momento de la evaluación el profesor revisa el examen y le da la nota al estudiante, si este está conforme entonces se envía por correo electrónico o personalmente al responsable de los cursos, de lo contrario, no se toma en cuenta la nota del estudiante. Posteriormente esta nota es enviada a secretaría para conformar el acta y acreditarle el curso al estudiante, para esto se le dan los siguientes datos: nombre y apellidos, grupo, curso y nota final. Una vez realizada el acta, se firma la misma por parte del profesor o AA que imparte el curso, entregándose nuevamente a la secretaria, y luego es pasada la nota a Akademos.

Posteriormente se envía por correo electrónico a todos los estudiantes una convocatoria de los cursos optativos que se ofertarán, donde se explica como será el procedimiento de la matrícula y a quién estará dirigido cada curso (en dependencia del año que cursa, condiciones necesarias, si está en proyecto, etc.), se envía el horario, el que lo impartirá, la capacidad, el local y la fecha de inicio. Durante un plazo determinado el estudiante pasa con su solapín por la Secretaría y se matricula en un curso según el que desee y las condiciones que debe cumplir para esto, o se le envía directamente el curso que recibirá según el grupo o si está vinculado a proyecto, todo depende de la necesidad de la facultad y la estrategia que se tome en este sentido. Una vez vencido el plazo (cuando es a convocatoria abierta), se conforman los grupos de cada curso y se le entrega o envía por vía electrónica al que lo impartirá, quien se encargaría conjuntamente con el responsable, de informar a sus estudiantes sobre todo lo referente al curso hasta su culminación.

Cuando el curso termina con el examen final o la última clase, el que lo imparte registra la nota de cada estudiante (depende del trabajo final y el recorrido) y la envía por correo electrónico al responsable, este archiva la nota en un Excel (donde lleva un control) y la envía a la Secretaría, ahí se elaboran las actas, luego se avisa a los profesores para firmarlas y posteriormente son pasadas a Akademos.

Aquel estudiante que por algún motivo termina con 2 puntos, simplemente no se le acredita el curso.

Si el curso es convocado a nivel central, todos los procesos a desarrollarse son independientes de la Facultad, solo se necesita recibir información de qué curso fue y los estudiantes involucrados, para tener un control más actualizado del estado de cada uno al respecto.

En el caso del perfil, en la facultad solo existe uno, que es el de Multimedia y Software Educativo, por lo que el estudiante no tiene para escoger, lo cual no quita que pueda pasar otro perfil fuera de la facultad, donde el requisito solo se enmarca en que debe tener al menos uno aprobado al graduarse. Para esto, solo debe recibir las asignaturas, tanto básicas (algunas se homologan con cursos optativos) como optativas (cursos optativos) que lo conforman, que son un número determinado en dependencia del perfil. Para organizar, planificar y controlar el perfil o los perfiles de la facultad, hay un responsable, que junto a los directivos de la Facultad, toma decisiones y traza estrategias para obtener un mayor éxito y lograr un proceso óptimo y con la calidad requerida.

Cuando se necesita información para sacar una estadística sobre el estado de los cursos o el perfil en la Facultad, hay que buscar estudiante a estudiante en Akademos e ir verificando en el Excel que tiene la persona responsable, o verificar las actas, luego establecer los parámetros para obtener los reportes determinados que se necesitan o se piden por parte de la Facultad, y todo esto se hace en un Excel, permitiendo filtrar información, hacer cálculos y controlar un poco la información para trazar estrategias y tomar decisiones. Todo este proceso toma su tiempo y no se puede entregar rápido cuando el plazo de petición de datos es muy corto.

Estos controles se hacen periódicamente y además, son necesarios antes de cada convocatoria de los cursos, ya que se necesita saber cuántos cursos ha pasado el estudiante y cuántos les falta para el cumplimiento del perfil, para luego dar prioridades a aquellos que están más críticos o que lo necesitan en mayor medida.

El negocio termina cuando la nota es pasada al acta, firmada y luego pasada a Akademos.

### 2.3. Descripción de los procesos del negocio.

En cualquier universidad nacional o extranjera se gestiona la información inherente a las materias académicas que recibe el estudiante mediante los diferentes procesos y estrategias que se lleven a cabo en la dirección general. La gestión de los cursos optativos en la Facultad #8 de la Universidad de las Ciencias Informáticas es el campo de acción, y sobre él se basa la construcción del modelo del negocio: reflejo de lo que desde el punto de vista real ocurre.

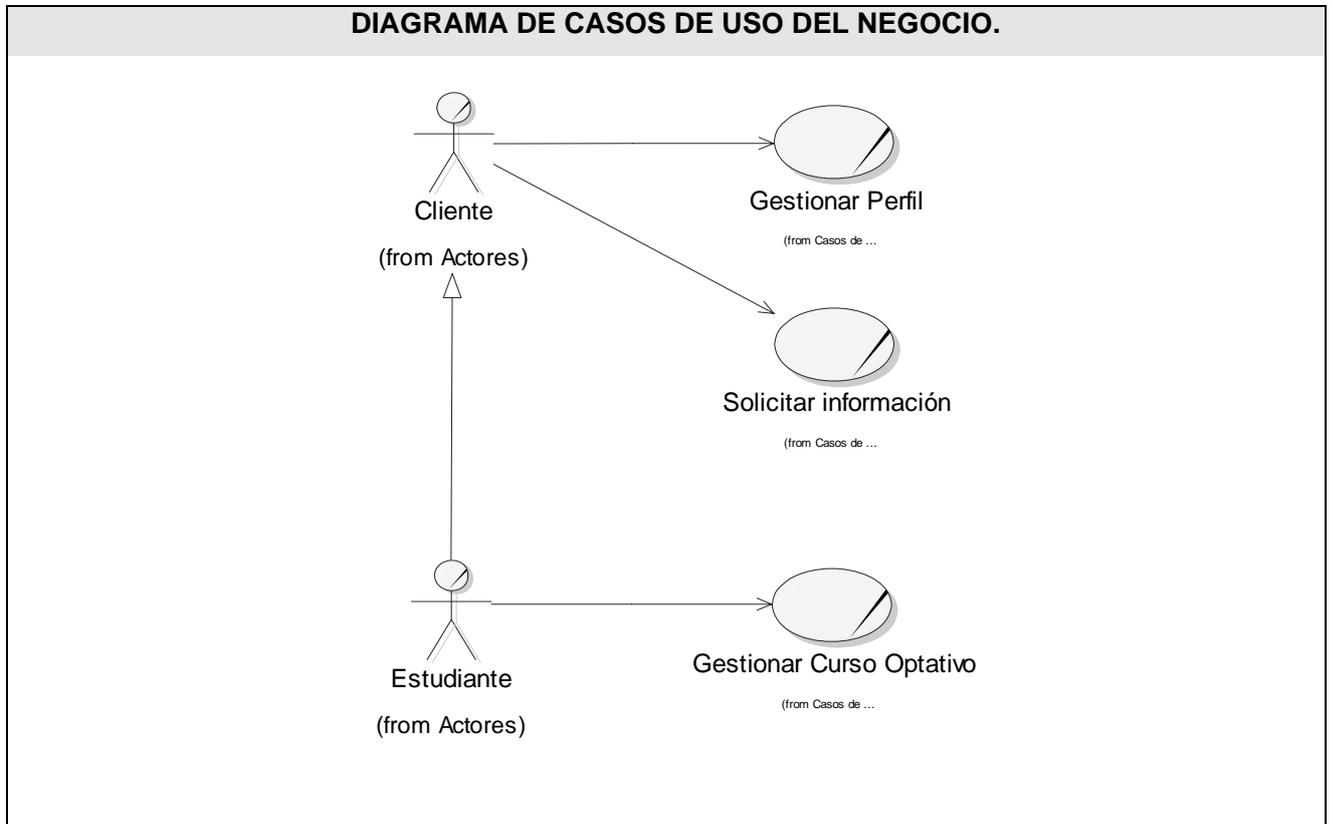
En dicho proceso participan diferentes actores y trabajadores del negocio que toman diferentes roles para lograr la gestión de toda la información y que interactúan con los Casos de Uso del Negocio (CNU).

**Table 1** Descripción de los actores del negocio.

Actores del negocio	Justificación
Estudiante	Interactúa con el negocio con el fin de solicitar curso optativo motivado por la necesidad de recibirlo o interés de la Facultad en algunos casos, así como obtener datos del estado de estos.
Interesado de la Facultad	Interactúa con el negocio con el fin de hacer alguna petición sobre determinada información de interés que le permita tomar decisiones o cumplir con cierta tarea asignada a nivel central. Además, da apoyo para un mayor éxito del perfil ofertado al estudiante.
Cliente	Es una generalización de los actores anteriores, que interactúa con el negocio para solicitar información, además de jugar diferentes roles en el proceso relacionado con la gestión del Perfil.

**Table 2** Descripción de los trabajadores del negocio.

<b>Trabajadores del negocio</b>	<b>Justificación</b>
Responsable de los Cursos Optativos	Encargado de definir, planificar, organizar y controlar los cursos optativos dentro de la Facultad.
Responsable del Perfil	Encargado de planificar, organizar y controlar el perfil dentro de la Facultad.
Secretaria docente	Encargada de matricular a los estudiantes, acreditar la nota del curso mediante las actas, pasarlas al sistema académico y llevar cierto control de esto.
Profesor	Profesor o AA encargado de impartir el curso, evaluar y firmar la nota en acta de cada estudiante.



**Fig 5** Diagrama de Casos de Uso del Negocio.

A continuación se presenta una descripción de cada CUN, donde se detallan los procesos que en estos se ejecutan mediante el diagrama de actividades.

**2.3.1. Gestionar Cursos Optativos.**

**Table 3** Descripción textual del CUN Gestionar Cursos Optativos.

<b>Caso de Uso del Negocio</b>	<b>Gestionar Cursos Optativos</b>
<b>Actores</b>	Estudiante
<b>Propósito</b>	Que los estudiantes puedan recibir y acreditar un curso optativo según su interés o necesidad, como resultado de la planificación y organización de estos por parte del responsable. Además, este proceso permitirá llevar un control del estado de cada estudiante en materia de cursos.
<b>Casos de uso Asociados</b>	<b>Ninguno</b>
<b>Resumen</b>	
El caso de uso se inicia cuando el estudiante, ante la petición por parte de la facultad o en su interés propio se presenta a convalidar o recibir un curso optativo de los que se ofertan. Luego que es aceptado para una u otra actividad, y culmina la misma satisfactoriamente, su nota es acreditada en acta, firmada y luego pasada a Akademos.	
<b>Flujo de Trabajo</b>	
<b>Acciones del actor</b>	<b>Respuesta del proceso de negocio</b>
<b>1)</b> El estudiante solicita cursos según necesidades de la Facultad o interés personal.	<b>2)</b> El responsable de los cursos se reúne con todos los alumnos ayudantes (AA) y profesores que van a impartir los cursos para definir, planificar y organizar los cursos optativos que se ofertarán.

	<b>3)</b> Se traza estrategia para controlar y gestionar los riesgos que puedan surgir durante el período de ofertas de cursos.
	<b>4)</b> Se propone un horario de conveniencia para cada AA o profesor que impartirá el curso
	<b>5)</b> Se envía por correo electrónico a todos los estudiantes una convocatoria de los cursos optativos que se ofertarán, donde se explica como será el procedimiento de la matrícula y a quién estará dirigido cada curso en dependencia del año que cursa, condiciones necesarias, si está en proyecto, etc.
	<b>6)</b> El responsable de los Cursos Optativos establece un plazo determinado para solicitar los cursos a examinar o recibir.
<b>7.1)</b> El estudiante se presenta en la secretaría para solicitar su curso óptativo.	<b>8)</b> La secretaria le pregunta qué curso desea y pide los datos: nombre, solapín y grupo.
<b>9)</b> El estudiante aporta los datos pedidos.	<b>10)</b> La secretaria incluye a este estudiante en la lista del curso que ha elegido.
	<b>11)</b> La secretaria le facilita a los profesores o AA que impartirán el curso los listados con los nombres de los estudiantes que lo han solicitado.
	<b>12)</b> El responsable de los cursos envían un correo a todos los matriculados y el profesor con el horario y el curso, así como la fecha de inicio.

13) El estudiante recibe el curso.	14) Los profesores, una vez concluido el curso, entregan los listados con las notas de los estudiantes al responsable del curso. Los datos a entregar son: nombre, grupo, nota y curso.
	15) El responsable de los cursos organiza los resultados y los envía por vía electrónica a la secretaria.
	16) La secretaria conforma las actas con el nombre, el grupo, el curso y las notas de los estudiantes.
	17) El profesor firma el acta creada por la secretaria.
18) El estudiante acredita el curso.	19) La secretaria pasa las notas de los estudiantes en Akademos.
<b>Cursos alternos:</b>	
<b>Acciones del actor</b>	<b>Respuesta del proceso de negocio</b>
7.2) El estudiante, una vez convocado el examen de suficiencia de los cursos ofertados, se presenta a la secretaría a solicitar esta acción o lo hace por correo.	8) La secretaria conforma y entrega los listados de los estudiantes que solicitaron los exámenes al responsable de los cursos.
	9) El responsable de los cursos envía un correo a los profesores que deben aplicar los exámenes y los estudiantes que lo solicitaron, con el objetivo de informar el horario y la fecha en que serán realizados.
	10) Los profesores aplican la prueba de suficiencia.
	11) Los profesores entregan al responsable de los cursos los listados con las notas de los

	estudiantes en el curso que convalidó.
	<b>12)</b> El responsable de los cursos organiza los resultados y los envía por vía electrónica a la secretaria.
	<b>13)</b> La secretaria conforma las actas con el nombre, el grupo, el curso y las notas de los estudiantes.
	<b>14)</b> El profesor firma el acta creada por la secretaria.
<b>15)</b> El estudiante acredita el curso.	<b>16)</b> La secretaria pasa las notas de los estudiantes en Akademos.
<b>Prioridad</b>	Crítica

**Gestionar Perfil.**

**Table 4** Descripción textual del CUN Gestionar Perfil.

<b>Caso de Uso del Negocio</b>	<b>Gestionar Perfil</b>
<b>Actores</b>	Cliente
<b>Propósito</b>	Que se planifique, organice y controle todo lo relacionado con los perfiles ofertados por la Facultad con sus asignaturas, de forma tal que el estudiante pueda recibir al menos uno para graduarse satisfactoriamente.
<b>Casos de uso Asociados</b>	-----

<b>Resumen</b>	
<p>El caso de uso comienza cuando se define por parte de los directivos de la facultad cuál o cuáles pueden ser los perfiles para complementar la especialidad de informática y se establecen las asignaturas básicas u optativas que ellos van a necesitar para podersele acreditar a los estudiantes, también se debe tener en cuenta que a nivel de universidad existen perfiles generales o en otra facultad, que en caso que algún estudiante opte por ellos debe registrarse en la facultad ese perfil, así como las asignaturas, para llevar el control del mismo. El responsable del perfil debe organizar y controlar toda la información referente al perfil.</p>	
<b>Flujo de Trabajo</b>	
<b>Acciones del actor</b>	<b>Respuesta del proceso de negocio</b>
<b>1</b> Se necesita a nivel de facultad (por parte de los directivos interesados) que los estudiantes, una vez lleguen a su año de graduación se les pueda incluir un perfil como complemento de la carrera de Ingeniería en Informática.	<b>2</b> Los responsables de este proceso definen los perfiles para la facultad con las asignaturas que el estudiante debe vencer para obtener el perfil.
<b>3</b> El estudiante solicita que se le incluya en un perfil determinado de la facultad, de otra facultad, o a nivel de Universidad.	
	<b>4</b> Se incluye al estudiante en un perfil determinado.
	<b>5</b> A los estudiantes que solicitaron un perfil de la facultad se les planifica un horario de clases de acuerdo a las asignaturas que necesite dicho perfil.
<b>6</b> Debe recibir todas las asignaturas correspondientes.	<b>7</b> Se les debe ir acreditando las asignaturas a los estudiantes y al concluir ya tendrá acreditado su perfil.
<b>Prioridad</b>	Crítica

**Solicitar información.**

**Table 5** Descripción textual del CUN Solicitar Información.

<b>Caso de Uso del Negocio</b>	<b>Solicitar información</b>
<b>Actores del Negocio</b>	Cliente
<b>Propósito</b>	El propósito es que el cliente pueda pedir información al negocio para realizar algún reporte, estadística o informe acerca del estado de los cursos optativos y/o el Perfil.
<b>Casos de uso Asociados</b>	<b>Ninguno</b>
<b>Resumen</b> El caso de uso se inicia cuando algún cliente de la Facultad solicita alguna información, estadística o reporte sobre los cursos optativos o el Perfil. Los trabajadores del negocio deben de procesar la información que presentan, con el objetivo de ofrecer al cliente lo solicitado. El caso de uso termina cuando este último recibe la información.	
<b>Flujo de Trabajo</b>	
<b>Acciones del actor</b>	<b>Respuesta del proceso de negocio</b>
<b>1</b> El Cliente solicita una información, reporte o estadística del estado de los cursos optativos o el Perfil.	<b>2</b> Los trabajadores del negocio procesan la información que poseen según las necesidades del cliente.
<b>3</b> El Cliente recibe la información, reporte o estadística.	
<b>Prioridad</b>	Crítica
<b>Cursos alternos:</b> -----	

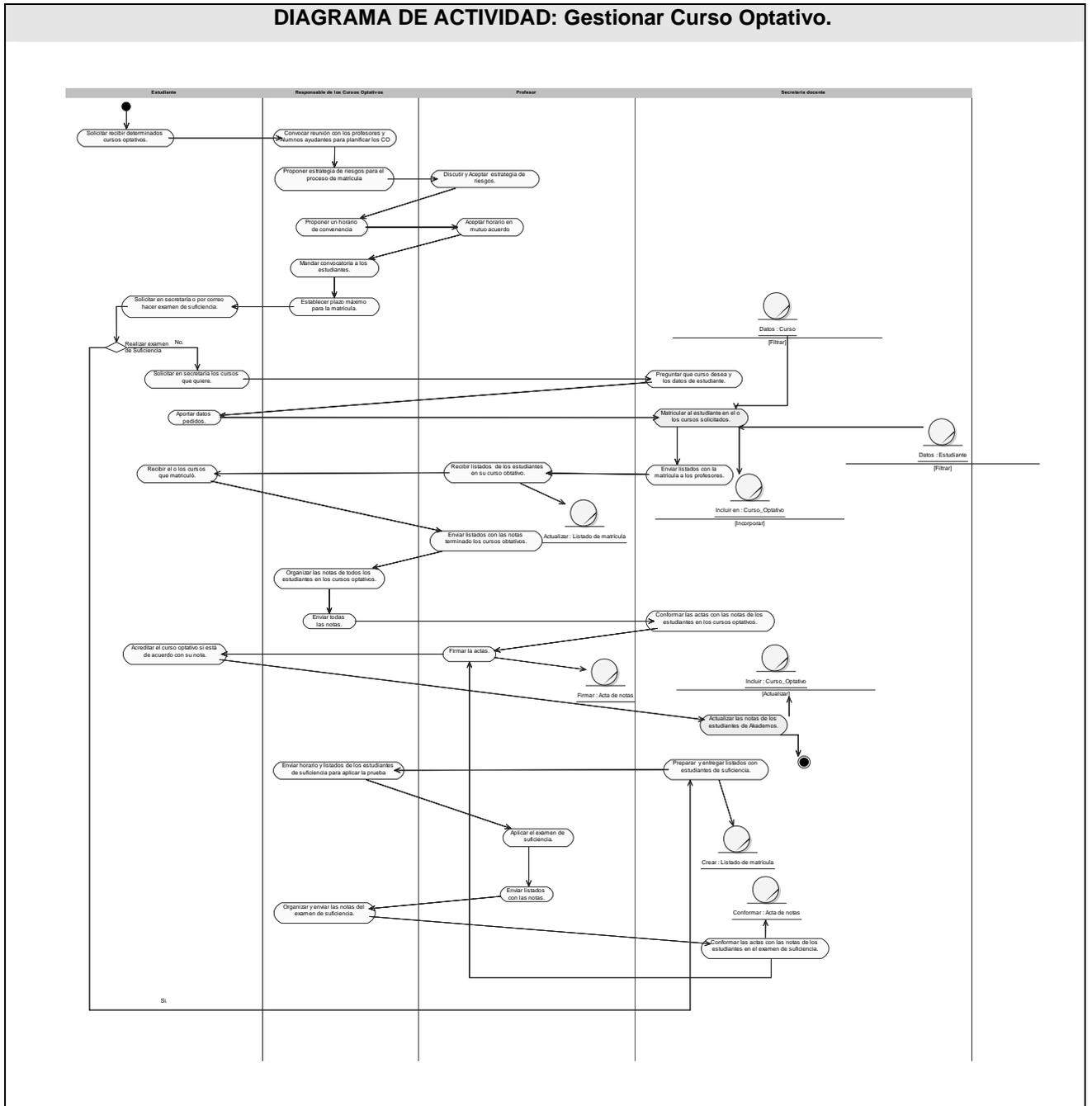


Fig 6 Diagrama de Actividades del CUN Gestionar Curso Optativo.

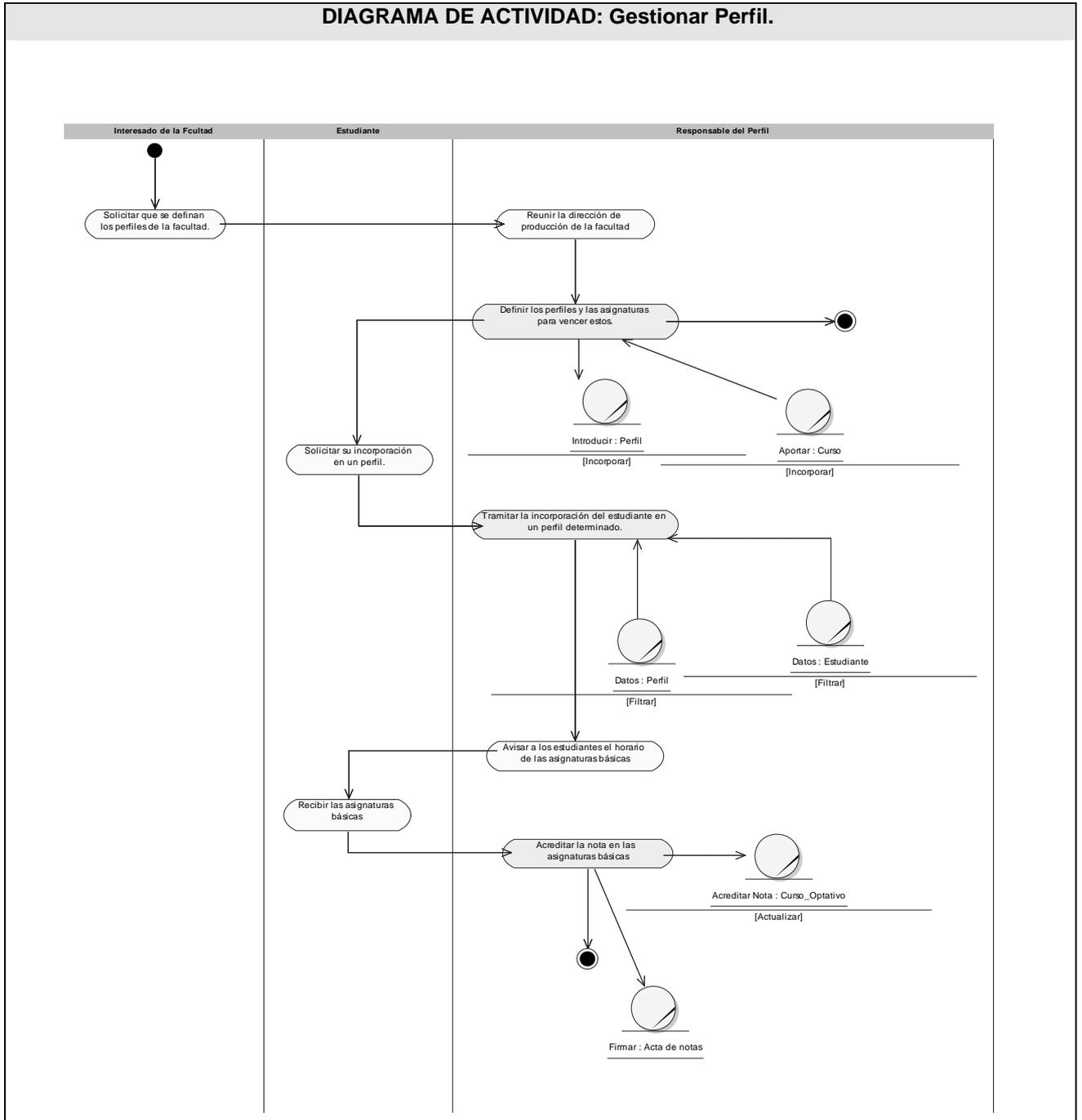
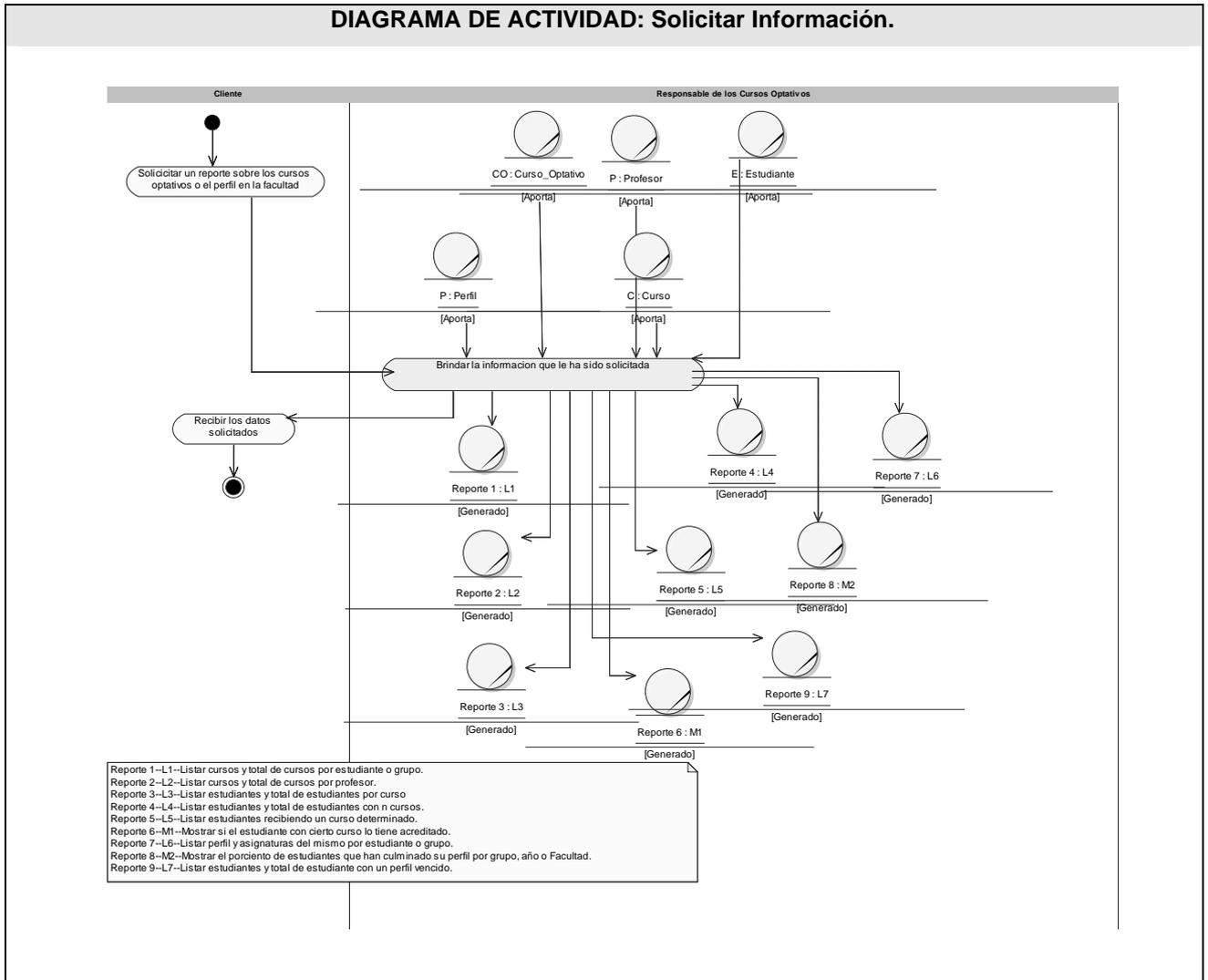


Fig 7 Diagrama de Actividades del CUN Gestionar Perfil.



**DIAGRAMA DE CLASES DEL MODELO DE OBJETOS**

**Fig 8** Diagrama de Actividades del CUN Solicitar Información.

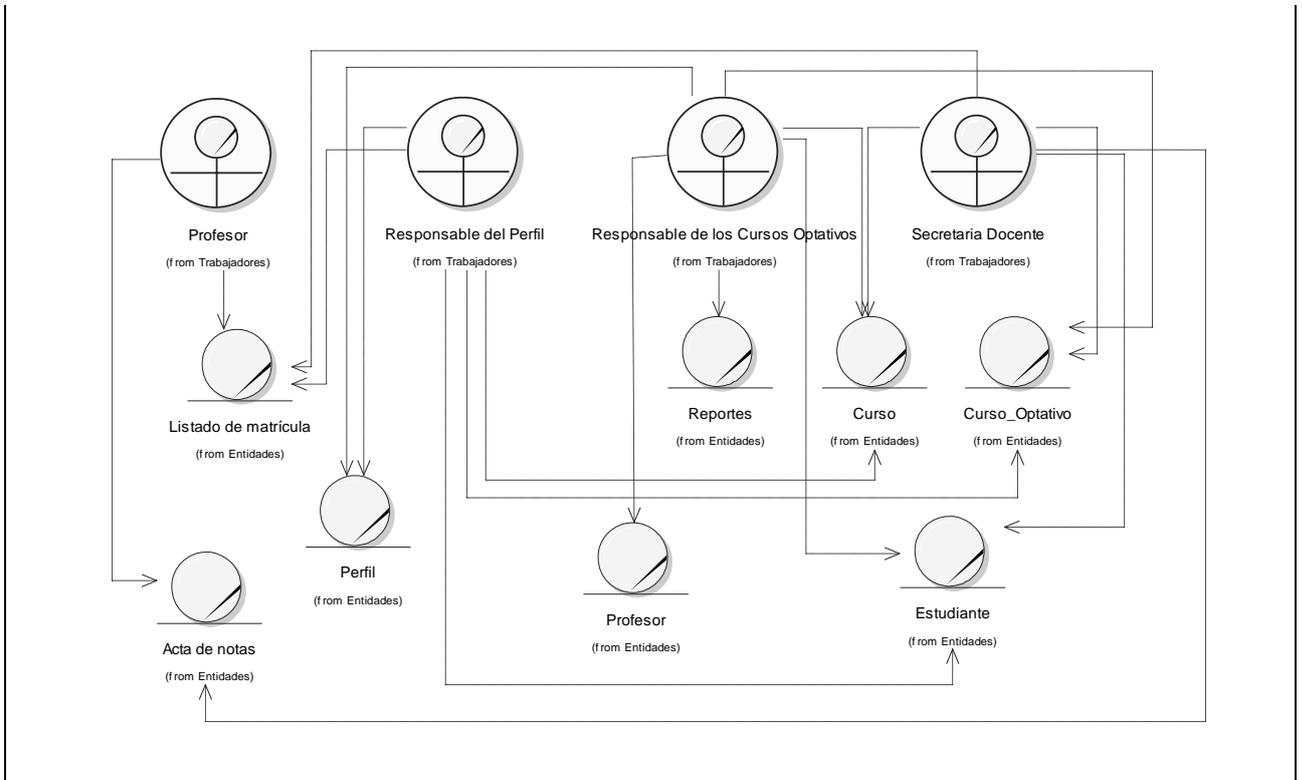


Fig 9 Modelo de Objetos del negocio.

## 2.4. Requerimientos funcionales.

### RF 1 Autenticar Usuario.

### RF 2 Gestionar curso.

- 2.1. Actualizar curso.
- 2.2. Modificar curso.
- 2.3. Listar los cursos.
- 2.4. Mostrar los detalles de un curso.
- 2.5. Mostrar cantidad de cursos.

**RF 3 Gestionar perfil.**

- 3.1. Insertar perfil.
- 3.2. Modificar perfil.
- 3.3. Eliminar perfil.
- 3.4. Listar perfil.
- 3.5. Mostrar cantidad de perfiles.
- 3.6. Mostrar los detalles de un perfil.

**RF 4 Gestionar información.**

- 4.1. Listar cursos y total de cursos por estudiante.
- 4.2. Listar cursos y total de cursos por grupo.
- 4.3. Listar cursos y total de cursos por profesor.
- 4.4. Listar estudiantes y total de estudiantes por curso.
- 4.5. Listar estudiantes y total de estudiantes con *n* cursos.
- 4.6. Listar estudiantes recibiendo un curso determinado.
- 4.7. Mostrar si el estudiante con cierto curso lo tiene acreditado.
- 4.8. Listar perfil y asignaturas del mismo por estudiante.
- 4.9. Listar perfil y asignaturas del mismo por grupo.
- 4.10. Mostrar el por ciento de estudiantes que han culminado su perfil por grupo.
- 4.11. Mostrar el por ciento de estudiantes que han culminado su perfil por año.
- 4.12. Mostrar el por ciento de estudiantes que han culminado su perfil por facultad.
- 4.13. Listar estudiantes y total de estudiantes con un perfil vencido.

**RF 5 Gestionar matrícula.**

Matricular en un curso optativo.

Matricular en un perfil.

Vaciar la capacidad del curso optativo una vez terminado para volverlo a ofertar.

## **2.5. Requerimientos no funcionales**

### **Definición de los requisitos no funcionales.**

Los requerimientos no funcionales son características que describen alguna forma o restricción para la realización de algún requerimiento (funcionalidad) o conjunto de ellas e inclusive todos los requerimientos. Se consideran los atributos del sistema, propiedades que debe tener el producto.

A continuación se muestran los requerimientos no funcionales:

### **RNF 1 Apariencia o interfaz de usuario.**

- 1.1. La interfaz a utilizar será una estándar escogida por el grupo de desarrollo, con cambios en los estilos, de forma que quede sencilla y atractiva.
- 1.2. Los colores escogidos deben de identificar a nuestra universidad.
- 1.3. Debe ser interactiva.

### **RNF 2 Usabilidad.**

- 2.1. El sistema será diseñado para que cualquier persona que tenga algún conocimiento básico de computación y de trabajo en la Web pueda utilizarlo.
- 2.2. El tamaño del texto debe ser promedio para que pueda ser observado a un metro o más de distancia.

### **RNF 3 Rendimiento.**

- 3.1. El sistema debe gestionar toda la información y dar respuesta a las solicitudes lo más rápido y eficientemente posible.
- 3.2. Debe estar disponible en todo momento.

### **RNF 4 Soporte.**

- 4.1. Se deben realizar varias pruebas demostrando confiabilidad y precisión.
- 4.2. Debe ser de fácil mantenimiento y de configuración sencilla.

**RNF 5 Portabilidad.**

5.1. Multiplataforma. El sistema se podrá montar sobre Unix, Linux (Nescape Navigator), Windows (Internet Explorer 6).

**RNF 6 Seguridad.**

6.1. Debe permitir al usuario autenticarse, permitiéndole realizar las acciones correspondientes según los permisos.

**RNF 7 Políticos-culturales.**

7.1. Interfaz que denote el lugar donde se encuentra a través de imágenes, frases, etc.

**RNF 8 Legales.**

8.1. Reconocido y autorizado por instancias superiores tales como la directiva de la Facultad 8 y de la UCI.

8.2. Documentación legal de uso como Declaración de Autoría.

**RNF 9 Confiabilidad.**

9.1. Disponible en todo momento debido a la importancia de la información que gestiona.

9.2. Antes de cualquier desperfecto funcional se dará a conocer enseguida a la comunidad de usuarios y se tratará de resolver en el mismo día. Si es problema de mantenimiento se tratará de que sea en días no laborables. Una vez terminada cualquiera de las actividades anteriores, se dará una notificación de terminación mostrando las disculpas por las afectaciones causadas.

9.3. Los reportes y estadísticas que se obtendrán deben ser 100% precisos y reales.

9.4. Capaz de restaurarse de las fallas rápidamente.

**RNF 10 Ayuda y documentación en línea.**

10.1. Documentación de ayuda para uso del sistema y reportes solicitados que estarán asequibles desde cualquier parte del sistema siempre y cuando esté autenticado el usuario.

**RNF 11 Software.**

- 11.1. Para el servidor Web: Servidor Web Apache v1.x o superior.
- 11.2. SGDB: PostgreSQL v7.x o superior.

**RNF 12 Hardware.**

- 12.1. Tarjeta de red.
- 12.2. Para los servidores tanto Web como SGBD: 256 RAM.
- 12.3. Capacidad de disco duro en Gigabyte, preferiblemente mayor a los 10 GB.
- 12.4. PC a 1GHz de Velocidad del Microprocesador o superior.

**RNF 13 Restricciones en el diseño y la implementación.**

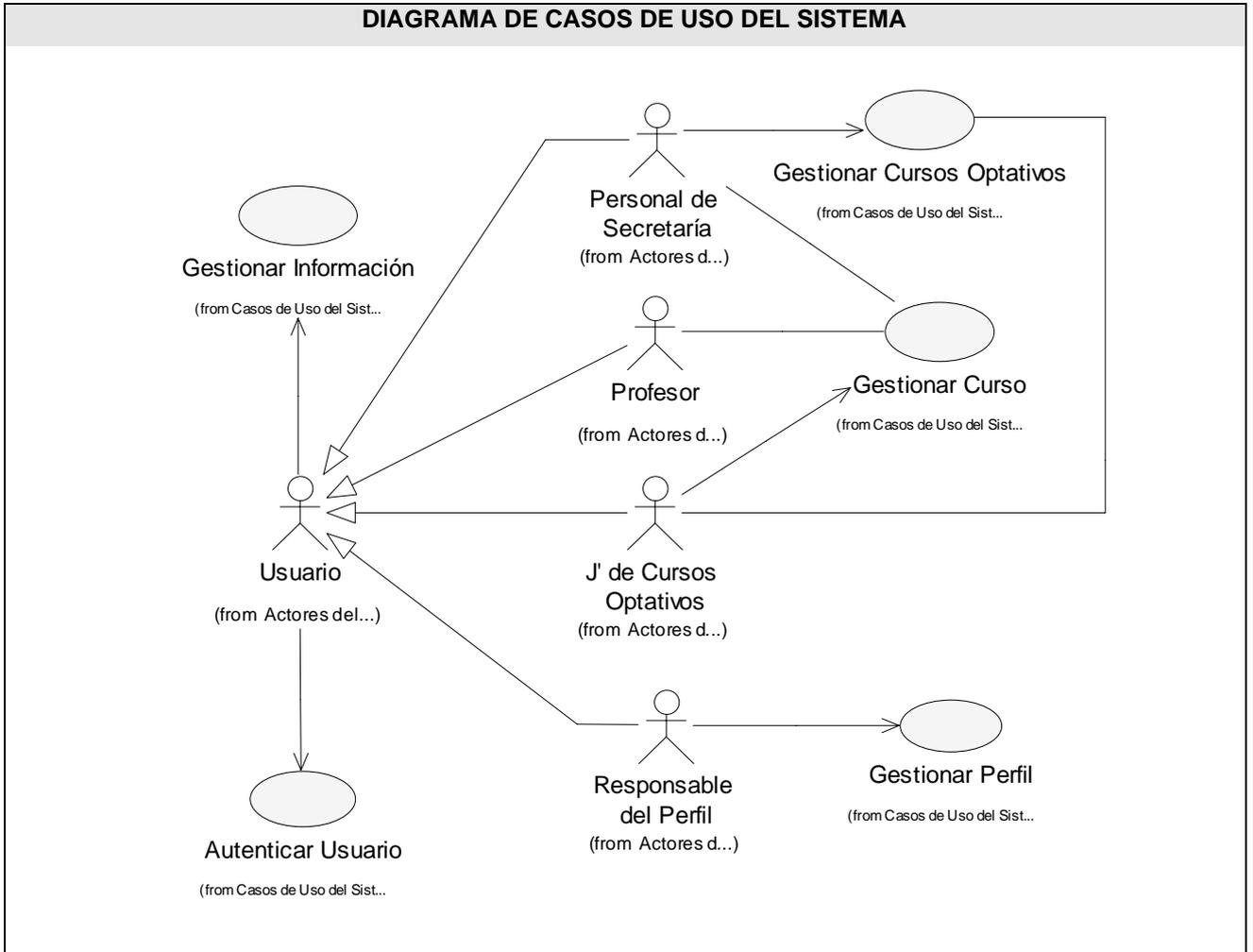
- 13.1. Para el diseño se requiere del uso de Rational Rose pues la metodología usada es RUP a través del lenguaje UML.
- 13.2. Para la programación en PHP se recomienda el NuSphere PhpED.
- 13.3. Para servidor de gestión de datos se requiere de PostgreSQL por ser un software de licencia gratis o reconocido como software libre.

**2.6. Definición de los Casos de uso del sistema**

**Table 6** Descripción de los actores del negocio.

Actores del negocio	Justificación
Profesor	Interactúa con el sistema con el objetivo de ver el listado de los estudiantes que recibirán su curso y poder modificar la nota final al culminar.
Personal de Secretaría	Interactúa con el sistema con el objetivo de matricular al estudiante en uno o varios cursos optativos, así como en el perfil que desee aprobar, acreditando su nota una vez culminada

	satisfactoriamente cada tarea.
J' de Cursos Optativos	Interactúa con el sistema con el objetivo de gestionar los cursos optativos a ofertar y llevar un control de estos.
J' del Perfil	Interactúa con el sistema con el objetivo de definir, planificar y organizar el perfil, y llevar el control de este.
Directivos	Interactúa con el sistema para definir el perfil que pasará cada estudiante.
Usuario	Es una generalización de los actores anteriores, que interactúa con el sistema para pedir y procesar información inherente a los cursos, perfil y estado de cada estudiante en relación a estos.
Responsable del Perfil	Es una generalización de los actores J' del Perfil y Directivos, que tiene como rol interactuar con el sistema para gestionar la información relacionada con el Perfil, dígame tipo de perfil, sus asignaturas, notas una vez culminada cada una de estas, etc.



**Fig 10** Diagrama de Casos de Uso del Sistema.

## Descripción de los Casos de Uso del Sistema

**Table 7** Descripción textual del CUS Autenticar Usuario.

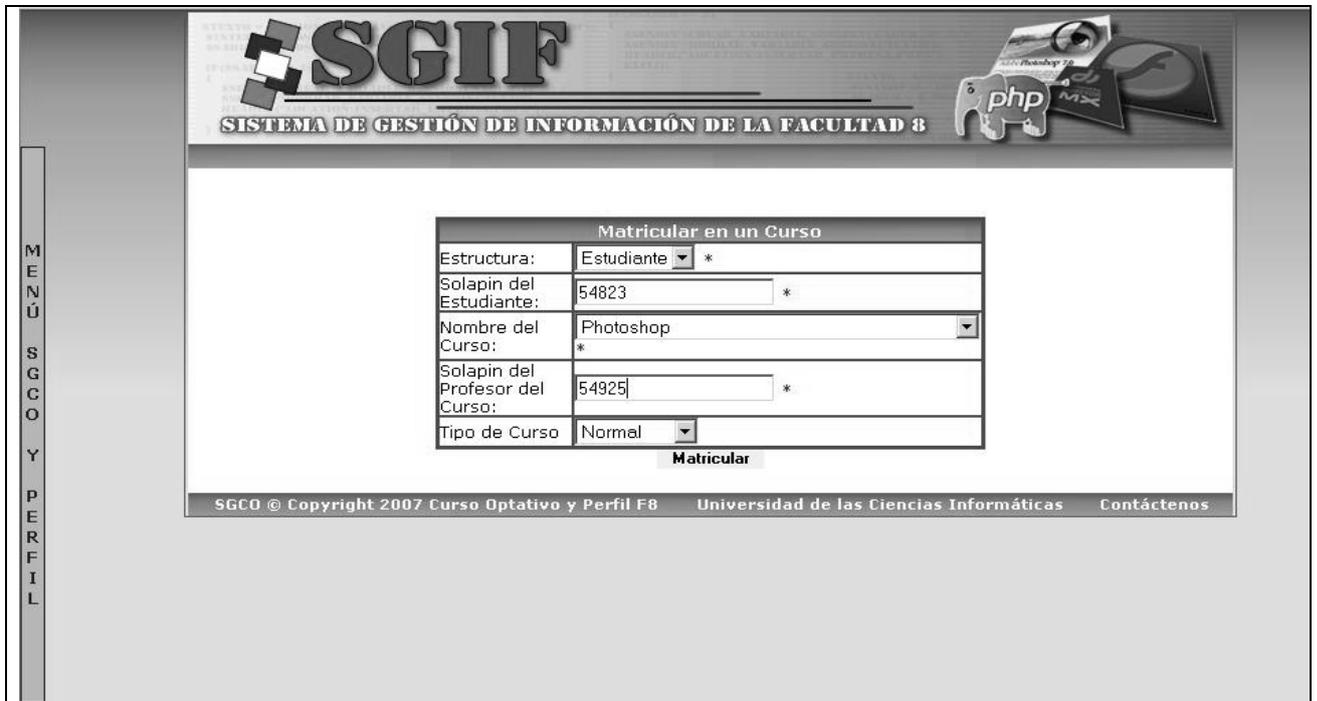
<b>Nombre del Caso de uso: Autenticar Usuario.</b>			
<b>Actores</b>		Usuario	
<b>Propósito.</b>		Posibilitar a los usuarios identificarse en el sistema para acceder a las opciones disponibles para su nivel de acceso.	
<b>Resumen:</b> El caso de uso permite que los usuarios con determinadas responsabilidades en el sistema introduzcan sus credenciales (usuario y contraseña) locales para que el sistema las verifique y ejecuten las funcionalidades que puedan según su rol.			
<b>Referencias</b>		<b>RF 1</b>	
<b>Curso Normal de los Eventos</b>			
<b>Acción del Actor</b>		<b>Respuesta del proceso del Sistema</b>	
<b>1</b>	El actor entra a la página principal del sistema.	<b>1.1</b>	El sistema muestra la opción para que el actor se autentique.
<b>2</b>	El actor introduce sus datos.	<b>2.1</b>	El sistema verifica la identidad del usuario dándole acceso de esta manera a las funcionalidades que le fueron definidas.
<b>Curso Alternativo de los eventos</b>			
<b>Acción 2.1:</b>		Si los datos introducidos son incorrectos el sistema muestra un mensaje de error y obliga retornar a la <b>acción 2</b> .	
<b>Prototipo de Interfaz.</b>			

<b>Prioridad:</b>	Crítico
-------------------	---------

**Table 8** Descripción textual del CUS Gestionar Cursos Optativos.

<b>Nombre del Caso de uso: Gestionar Cursos Optativos.</b>	
<b>Actores</b>	Personal de Secretaría
<b>Propósito.</b>	Matricular a los estudiantes en los cursos optativos ya sea por suficiencia, o por curso normal. Además se le puede dar baja a los estudiantes que una vez terminado el curso no hayan vencido el mismo.
<b>Resumen</b>	
El caso de uso se inicia cuando la secretaria docente, ante la presencia del estudiante, o la petición de un interesado procede a matricular al mismo de acuerdo al interés del estudiante.	

<b>Referencias</b>		<b>RF 5.1, 5.3.</b>	
<b>Curso Normal de los Eventos</b>			
<b>Acción del Actor</b>		<b>Respuesta del proceso del Sistema</b>	
<b>1</b>	El actor se dirige al menú de navegación en la parte izquierda de la aplicación.	<b>1.1</b>	El sistema muestra todas las opciones que permite el software incluida la de Matricular en Curso y Eliminar Cursos con Nota 0 en la sección <b>Cursos optativos</b> .
<b>Escenario 1: Matricular en un curso optativo.</b>			
<b>1</b>	El actor se dirige al menú en la parte izquierda de la aplicación y selecciona la opción <b>Matricular en Curso</b> .	<b>1.1</b>	El sistema muestra una interfaz, la cual permite el proceso de matricular a un estudiante o un grupo.
<b>2</b>	El personal de secretaría introduce los datos del estudiante que será matriculado.	<b>2.1</b>	El sistema matricula al estudiante en un curso determinado con un profesor y la forma en que este pasará el curso (normal ó suficiencia).
<b>Prototipo de Interfaz.</b>			



**Escenario 2: Vaciar la capacidad del curso optativo una vez terminado para volverlo a ofertar.**

1	El personal de secretaría se dirige al menú en la parte izquierda de la aplicación y selecciona la opción <b>Eliminar Cursos con Nota 0.</b>	1.1	El sistema muestra una interfaz que permite escoger un curso determinado y eliminar todos los estudiantes matriculados que no recibieron nota al culminar el curso.
---	--	-----	---

**Curso Alterno de los Eventos.**

**Escenario 1: Matricular en un curso optativo.**

Acción del Actor		Respuesta del proceso del Sistema	
1	El personal de secretaría se dirige al menú en la parte izquierda de la aplicación y selecciona en la sección <b>Cursos optativos</b> la opción <b>Matricular en Curso.</b>	1.1	El sistema muestra una interfaz la cual permite el proceso de matrícula a un grupo.

<b>2</b>	El personal de secretaría escoge el grupo.	<b>2.1</b>	El sistema matricula al grupo en un curso determinado con un profesor.
<b>Prioridad</b>		Crítico.	

**Table 9** Descripción textual del CUS Gestionar Curso.

<b>Nombre del Caso de Uso: Gestionar Curso.</b>			
<b>Actores</b>		J' de Cursos Optativos, Profesor, Personal de secretaría.	
<b>Propósito</b>		Actualizar los datos referentes a los cursos.	
<b>Resumen</b> El caso de uso se inicia cuando el Jefe de los Cursos Optativos gestiona la información referente a los cursos, permitiendo actualizarlos y listarlos.			
<b>Referencias</b>		RF 2	
<b>Curso Normal de los Eventos</b>			
<b>Acción del actor:</b>		<b>Respuesta del proceso del Sistema</b>	
<b>1</b>	El actor se dirige al menú de navegación, en la parte izquierda de la aplicación.	<b>1.1</b>	El sistema muestra en la sección Curso las opciones: Actualizar Curso y Listado de Cursos.
<b>Escenario 1: Actualizar Curso.</b>			
<b>1</b>	El actor selecciona la opción <b>Actualizar Curso</b> .	<b>1.1</b>	El sistema muestra una interfaz para que el actor realice el proceso de actualización del curso que consiste en agregar información relevante para la facultad con respecto a los cursos.
<b>2</b>	El actor luego de escoger el curso y agregar la información presiona el botón	<b>2.1</b>	El sistema realiza el proceso de actualización.

	<b>Actualizar.</b>		
--	--------------------	--	--

**Curso Alterno de los eventos**

<b>Acción 2.1:</b>	Si los datos introducidos son incorrectos el sistema muestra un mensaje de error y obliga retornar a la <b>acción 2.</b>
--------------------	--

**Prototipo de Interfaz.**



**Escenario 2: Modificar Curso Optativo.**

<b>1</b>	El actor selecciona la opción <b>Listado de Cursos.</b>	<b>1.1</b>	El sistema muestra una interfaz con el listado de los cursos que existen actualmente registrados, y en la parte derecha de los datos de los cursos aparece una opción Modificar, la cual permite modificar los datos del curso.
----------	---	------------	---

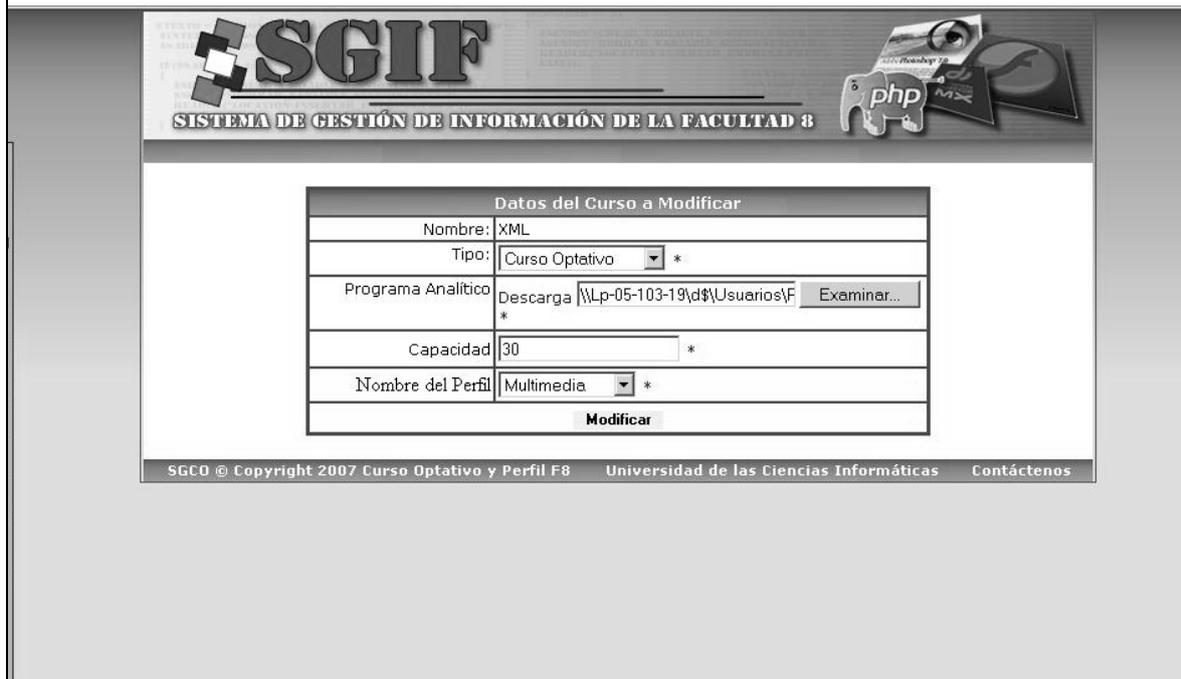
2	El actor elige el curso a modificar, para lo cual presiona la opción <b>Modificar</b> .	2.1	El sistema muestra una interfaz con los datos del curso para que el actor modifique el que desea.
3	El actor modifica y presiona el botón <b>Aceptar</b> para que se actualicen los cambios.	3.1	El sistema actualiza la información del curso modificado.

**Curso Alterno de los eventos**

**Acción 3.1:**

Si los datos introducidos son incorrectos el sistema muestra un mensaje de error y obliga retornar a la **acción 3**.

**Prototipo de Interfaz.**



**Escenario: 3 Listar cursos y Mostrar cantidad de cursos**

1	El actor selecciona la opción <b>Listado de Cursos</b> .	1.1	El sistema muestra una interfaz con el listado de los cursos y en la parte superior la cantidad de cursos existentes.
---	--	-----	---

**Prototipo de Interfaz.**



**Escenario: 4 Mostrar los detalles de un curso.**

1	El actor selecciona la opción <b>Listado de Cursos</b> .	1.1	El sistema muestra una interfaz con el listado de los cursos que existen actualmente registrados, y en la parte derecha de los datos de los cursos aparece la opción Detalles, para ver la información de un curso determinado.
2	El actor elige el curso para ver sus detalles, presionando la opción <b>Detalles</b> .	2.1	El sistema muestra una interfaz con la información del curso seleccionado.

**Prototipo de Interfaz.**

													
<b>Datos del Curso</b>	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th colspan="2" style="text-align: center;">Detalles del Curso</th> </tr> </thead> <tbody> <tr> <td style="text-align: right;">Nombre:</td> <td>HTML</td> </tr> <tr> <td style="text-align: right;">Capacidad:</td> <td>30</td> </tr> <tr> <td style="text-align: right;">Nombre del Perfil al que pertenece:</td> <td>Programación</td> </tr> <tr> <td style="text-align: right;">Programa Analítico:</td> <td style="text-align: center;">Descarga</td> </tr> <tr> <td style="text-align: right;">Asignatura Básica:</td> <td>NO</td> </tr> </tbody> </table> <p style="text-align: center; margin-top: 5px;"><a href="#">Volver</a></p>	Detalles del Curso		Nombre:	HTML	Capacidad:	30	Nombre del Perfil al que pertenece:	Programación	Programa Analítico:	Descarga	Asignatura Básica:	NO
Detalles del Curso													
Nombre:	HTML												
Capacidad:	30												
Nombre del Perfil al que pertenece:	Programación												
Programa Analítico:	Descarga												
Asignatura Básica:	NO												
SGCO © Copyright 2007 Curso Optativo y Perfil F8    Universidad de las Ciencias Informáticas    Contáctenos													
<b>Prioridad :</b>	Crítico.												

**Table 10** Descripción textual del CUS Gestionar Perfil.

<b>Nombre del Caso de Uso: Gestionar Perfil.</b>	
<b>Actores</b>	Responsable del Perfil
<b>Propósito</b>	Actualizar los datos referentes al Perfil.
<b>Resumen</b>	
El caso de uso se inicia cuando el Responsable del Perfil (J' del Perfil y Directivos) identifica los perfiles que se van a ofertar en la facultad, pudiendo actualizar su información.	
<b>Referencias</b>	RF 5.2, RF 3
<b>Curso Normal de los Eventos</b>	
<b>Acción del actor:</b>	<b>Respuesta del proceso del Sistema</b>

1	El actor se dirige al menú de navegación en la parte izquierda de la aplicación.	1.1	El sistema muestra en la sección <b>Perfil</b> del menú del software, las opciones: insertar perfil, modificar perfil, eliminar perfil, listado de perfiles y matricular en perfil.
---	--	-----	---

**Escenario 1: Insertar Perfil.**

1	El actor selecciona la opción <b>Insertar Perfil.</b>	1.1	El sistema muestra el formulario a completar para insertar un nuevo Perfil.
2	El actor introduce los datos y presiona el botón <b>Insertar.</b>	2.1	El sistema inserta el perfil con los datos introducidos.

**Curso Alternativo de los eventos**

<b>Acción 2.1:</b>	Si los datos introducidos son incorrectos, el sistema muestra un mensaje de error y obliga retornar a la <b>acción 2.</b>
--------------------	---

**Prototipo de Interfaz.**



**Escenario 2: Modificar Perfil.**

1	El actor selecciona la opción <b>Modificar Perfil.</b>	1.1	El sistema muestra una interfaz con la opción para escoger el perfil que se desea modificar.
2	El actor selecciona el perfil que desea modificar y presiona el botón <b>Aceptar.</b>	2.1	El sistema muestra una interfaz donde el actor puede modificar los datos.
3	El actor una vez modificado los datos presiona el botón <b>Modificar.</b>	3.1	El sistema modifica los datos.

**Curso Alterno de los eventos**

**Acción 3.1:**

Si los datos introducidos son incorrectos, el sistema muestra un mensaje de error y obliga retornar a la **acción 3.**

**Prototipo de Interfaz.**



**Escenario 3: Eliminar Perfil**

1	El actor selecciona la opción <b>Eliminar Perfil.</b>	1.1	El sistema muestra una interfaz en la que el actor puede escoger el perfil que desea
---	---	-----	--

			eliminar.
2	El actor selecciona el perfil que desea eliminar y presiona el botón <b>Eliminar</b> .	2.1	El sistema muestra un mensaje de confirmación por seguridad para que el actor verifique que en verdad desea borrar el perfil.
3	El actor presiona el botón <b>Aceptar</b> .	3.1	El sistema elimina el perfil y todas las asignaturas básicas de ese perfil pasan a ser cursos del estudiante.

**Curso Alterno de los eventos**

<b>Acción 3:</b>	El actor presiona el botón <b>Cancelar</b> y no se eliminan los datos.
------------------	--

**Prototipo de Interfaz.**

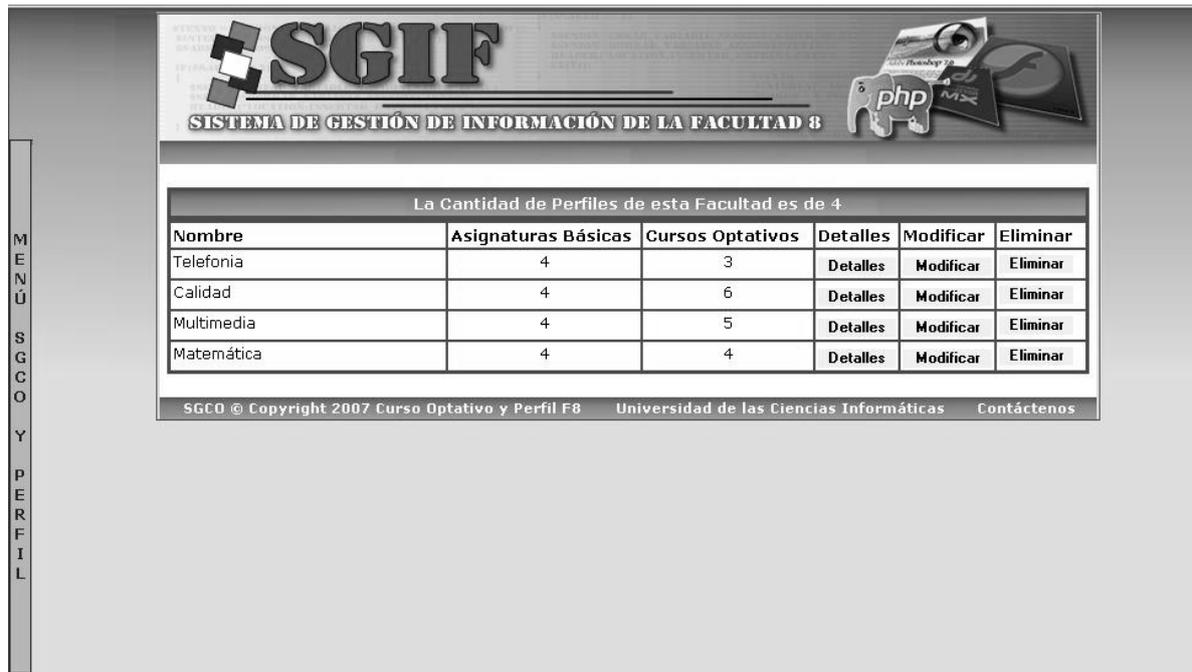


**Escenario: 4 Listar Perfil y Mostrar cantidad de perfiles.**

1	El actor selecciona la opción <b>Listado de Perfiles</b> .	1.1	El sistema muestra una interfaz con el listado de los perfiles que existen
---	--	-----	--

		actualmente registrados y con la cantidad que representan.
--	--	--

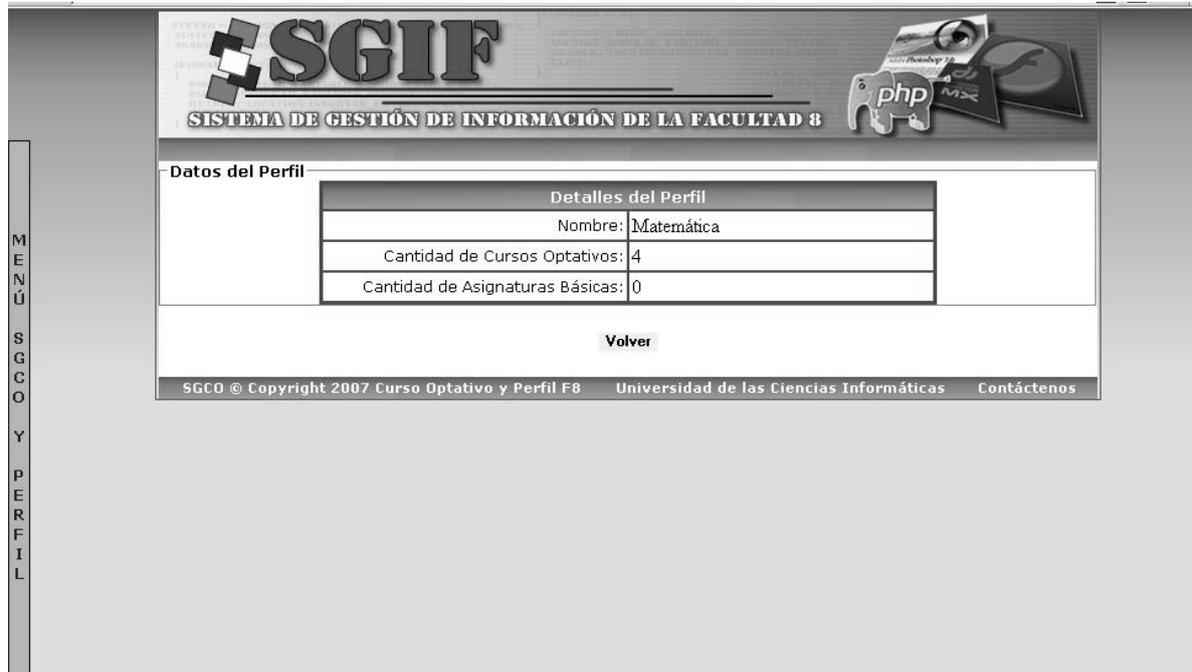
**Prototipo de Interfaz.**



**Escenario: 4 Permitir ver los detalles de un curso.**

<b>1</b>	El actor selecciona la opción <b>Listado de Perfiles</b> .	<b>1.1</b>	El sistema muestra una interfaz con el listado de los perfiles que existen actualmente registrados, y en la parte derecha de los datos de los cursos aparece una opción indicando que al ser presionada esta, la información de ese perfil podrá ser visualizada es decir (ver los detalles).
<b>2</b>	El actor escoge el perfil del cual quiere ver los detalles presionando el botón <b>Detalles</b> .	<b>2.1</b>	El sistema muestra una interfaz con la información del curso seleccionado.

**Prototipo de Interfaz.**



**Prioridad :** Crítico

**Table 11** Descripción textual del CUS Gestionar Información.

<b>Nombre del Caso de Uso: Gestionar Información.</b>	
<b>Actores</b>	Usuario
<b>Propósito</b>	Dar la posibilidad a los usuarios del sistema que accedan a los reportes que son solicitados con más frecuencia dentro y fuera de la Facultad respecto a los cursos optativos y los perfiles.
<b>Resumen</b>	
El caso de uso se inicia cuando un usuario del sistema necesita obtener algún reporte que este genera.	

<b>Referencias</b>		<b>RF 4</b>	
<b>Curso Normal de los Eventos</b>			
<b>Acción del actor:</b>		<b>Respuesta del proceso del Sistema</b>	
<b>1</b>	Se dirige hacia la parte izquierda de la aplicación donde se encuentra un menú desplegable con vínculos a todos los reportes que brinda.	<b>1.1</b>	<p>El sistema muestra el menú con las opciones:</p> <ol style="list-style-type: none"> <li>1. Listar cursos y total de cursos por estudiante.</li> <li>2. Listar cursos y total de cursos por grupo.</li> <li>3. Listar cursos y total de cursos por profesor.</li> <li>4. Listar estudiantes y total de estudiantes por curso.</li> <li>5. Listar estudiantes y total de estudiantes con <i>n</i> cursos.</li> <li>6. Listar estudiantes recibiendo un curso determinado.</li> <li>7. Mostrar si el estudiante con cierto curso lo tiene acreditado.</li> <li>8. Listar perfil y asignaturas del mismo por estudiante.</li> <li>9. Listar perfil y asignaturas del mismo por grupo.</li> <li>10. Mostrar el por ciento de estudiantes que han culminado su perfil por Grupo, Año ó Facultad.</li> <li>11. Listar estudiantes y total de estudiantes con un perfil vencido.</li> </ol>
<b>Escenario 1: Listar cursos y total de cursos por estudiante.</b>			

<b>1</b>	Selecciona la opción de <b>Cursos por Estudiante.</b>	<b>1.1</b>	El sistema muestra una interfaz para que el usuario busque al estudiante que desea, una vez que es identificado en la lista de posibles estudiantes.
<b>2</b>	Al presionar el botón <b>Seleccionar.</b>	<b>2.1</b>	El sistema muestra en otra interfaz los datos solicitados (Cursos por estudiante).
<b>Escenario 2: Listar cursos y total de cursos por grupo.</b>			
<b>1</b>	Selecciona la opción <b>Cursos por Grupo.</b>	<b>1.1</b>	El sistema muestra una interfaz para que el usuario escoja el grupo al que desea ver dichos datos.
<b>2</b>	Al presionar el botón <b>Aceptar.</b>	<b>2.1</b>	El sistema muestra en otra interfaz la lista con todos los nombres de estudiantes de dicho grupo
<b>3</b>	Al presionar el botón <b>Seleccionar</b> a la derecha del nombre del estudiante.		El sistema muestra en otra interfaz los datos solicitados (Cursos de los estudiantes de ese grupo).
<b>Escenario 3: Listar cursos y total de cursos por profesor.</b>			
<b>1</b>	Selecciona la opción <b>Cursos por Profesor.</b>	<b>1.1</b>	El sistema muestra una interfaz para que el usuario busque al profesor que desea, una vez que es identificado en lista de posibles profesores.
<b>2</b>	Al presionar el botón <b>Seleccionar.</b>	<b>2.1</b>	El sistema muestra en otra interfaz los datos solicitados (Cursos que imparte dicho profesor).
<b>Escenario 4: Listar estudiantes y total de estudiantes por curso.</b>			
<b>1</b>	Selecciona la opción <b>Estudiantes por Curso.</b>	<b>1.1</b>	El sistema muestra una interfaz para que el usuario escoja un curso determinado sobre el que desea saber cuántos

			estudiantes lo tienen acreditado.
<b>2</b>	Al presionar el botón <b>Aceptar</b> .	<b>2.1</b>	El sistema muestra en otra interfaz los datos solicitados (Una lista con los cursos que el estudiante tiene acreditado).
<b>Escenario 5: Listar estudiantes con n cursos.</b>			
<b>1</b>	Selecciona la opción <b>Estudiantes con n cursos</b> .	<b>1.1</b>	El sistema muestra una interfaz para que se entre la cantidad de cursos <b>n</b> .
<b>2</b>	Es entrada la cantidad y oprime el botón <b>Buscar</b> .	<b>2.1</b>	El sistema muestra en otra interfaz los datos solicitados (Una lista con los estudiantes que tienen n cursos acreditados).
<b>Escenario 6: Listar estudiantes recibiendo un curso determinado.</b>			
<b>1</b>	Selecciona la opción <b>Estudiantes recibiendo un curso</b> .	<b>1.1</b>	El sistema muestra una interfaz para que se entre el curso que desea analizar y el tipo de curso (normal o suficiencia).
<b>2</b>	Al presionar el botón <b>Aceptar</b> .	<b>2.1</b>	El sistema muestra en otra interfaz los datos solicitados (Una lista con los estudiantes que están pasando ese curso y el modo en que lo está recibiendo).
<b>Escenario 7: Mostrar si el estudiante con cierto curso lo tiene acreditado.</b>			
<b>1</b>	Selecciona la opción <b>Estudiante Acreditado Curso</b> .	<b>1.1</b>	El sistema muestra una interfaz para que el usuario busque al estudiante que desea, una vez que es identificado en la lista de posibles estudiantes.
<b>2</b>	Presiona el botón <b>Seleccionar</b> .	<b>2.1</b>	El sistema muestra una interfaz para que se escoja el curso a verificar.

<b>3</b>	Al presionar el botón <b>Aceptar</b> .	<b>3.1</b>	El sistema muestra un mensaje con el que indica si está o no acreditado el curso.
<b>Escenario 8: Listar perfil y asignaturas del mismo por estudiante.</b>			
<b>1</b>	Selecciona la opción <b>Perfil y Asignaturas por Estudiante</b> .	<b>1.1</b>	El sistema muestra una interfaz para que el usuario busque al estudiante que desea, una vez que es identificado en la lista de posibles estudiantes.
<b>2</b>	Al presionar el botón <b>Seleccionar</b> .	<b>2.1</b>	El sistema muestra en otra interfaz los datos solicitados (las asignaturas y el perfil del estudiante seleccionado).
<b>Escenario 9: Listar perfil y asignaturas del mismo por grupo.</b>			
<b>1</b>	Selecciona la opción <b>Perfil y Asignaturas por Grupo</b> .	<b>1.1</b>	El sistema muestra una interfaz para que el usuario escoja el grupo al que desea ver dichos datos.
<b>2</b>	Al presionar el botón <b>Aceptar</b> .	<b>2.1</b>	El sistema muestra en otra interfaz la lista con todos los nombres de estudiantes de dicho grupo.
<b>3</b>	Al presionar el botón <b>Seleccionar</b> .		El sistema muestra en otra interfaz los datos solicitados (las asignaturas y el perfil del estudiante seleccionado).
<b>Escenario 10: Mostrar el por ciento de estudiantes que han culminado su perfil por Grupo, Año ó Facultad.</b>			
<b>1</b>	Selecciona la opción <b>Mostrar Porcientos</b> .	<b>1.1</b>	El sistema da la posibilidad de seleccionar el modo de buscar: grupo, año ó facultad. Si escoje grupo muestra los grupos de la Facultad, si escoje año muestra los años (1ro, 2do, 3ro, 4to, 5to).

2	Escoge la opción que desea y presiona el botón <b>Aceptar</b> .	2.1	El sistema muestra el porciento solicitado en la misma interfaz.
<b>Escenario 11: Listar estudiantes y total de estudiantes con un perfil vencido.</b>			
1	Selecciona la opción <b>Estudiantes con Perfil Vencido</b>	1.1	El sistema muestra una interfaz para que el usuario escoja el perfil sobre el cual desea listar los datos.
2	Al presionar el botón <b>Aceptar</b> .	2.1	El sistema muestra los datos (la lista de estudiantes con el perfil vencido y la cantidad)
<b>Prioridad :</b>		Crítico	

Interfaz de una de la secciones.

ESTUDIANTES CON 2 CURSOS

Cantidad de Estudiantes: 12

Descargar Reporte

Foto:	Nombre del Estudiante:	Solapín	Grupo
	Dianelys Espinosa Zaldivar	26010	8501

>>

Página 1 de 12

Volver

SGCO © Copyright 2007 Curso Optativo y Perfil F8 Universidad de las Ciencias Informáticas Contáctenos

## **2.7. Conclusiones**

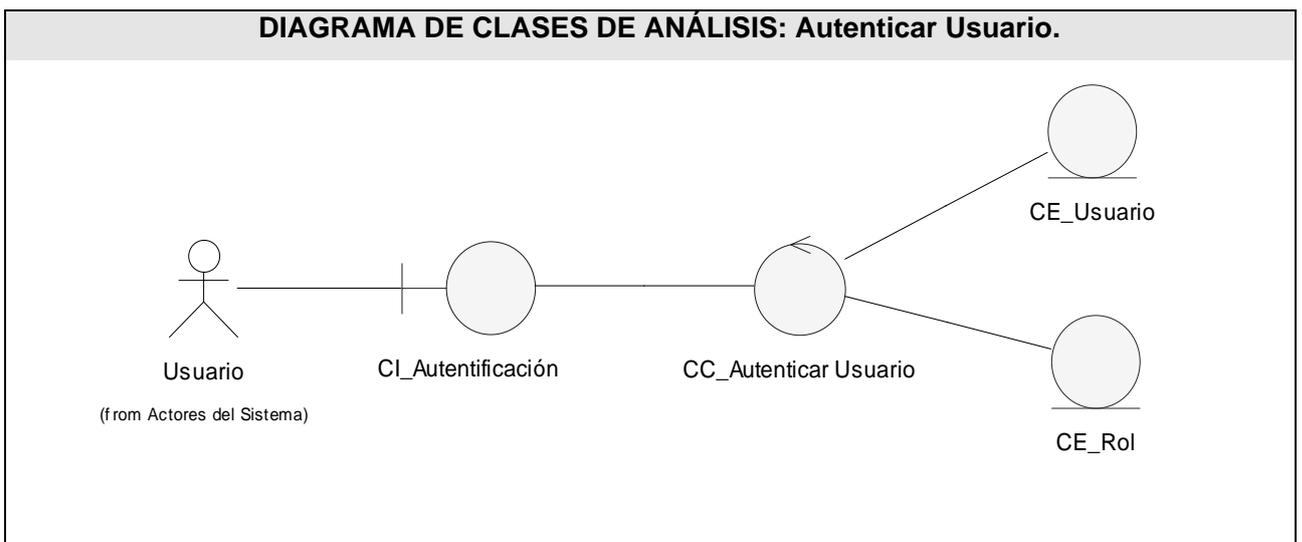
En este capítulo se han descrito todas las consideraciones necesarias para la propuesta de solución, constituyendo el punto de partida o los primeros pasos para el desarrollo del producto. A partir de los procesos de negocio se obtuvo un conjunto de funcionalidades que deben realizar el sistema, representados mediante casos de uso, y finalmente se describieron las acciones o flujo de información que desarrollarán los actores en el sistema. Además, se definieron un conjunto de requerimientos no funcionales que debe cumplirse para una construcción del sistema con más calidad y bien refinada.

## CAPÍTULO 3 ANÁLISIS Y DISEÑO DEL SISTEMA

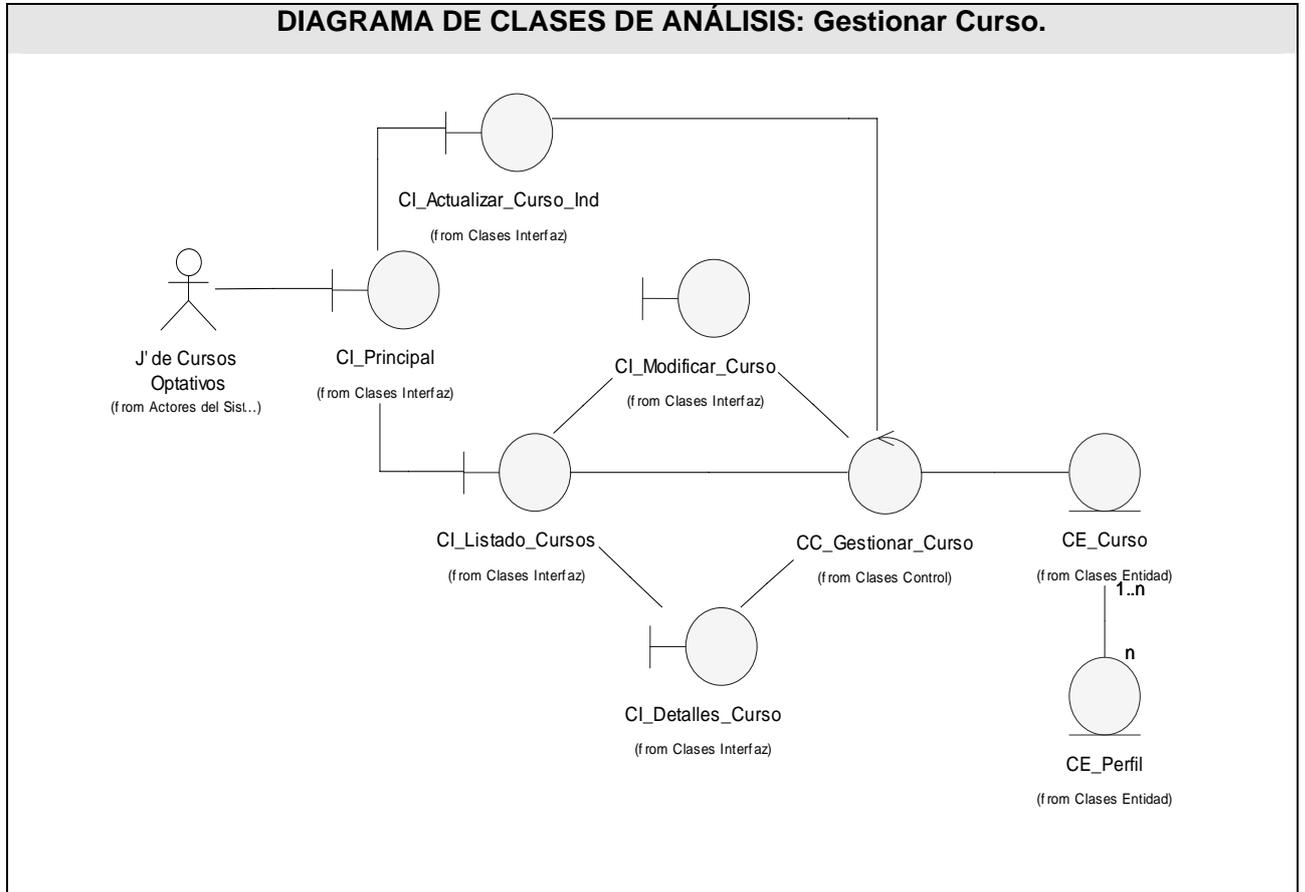
### 3.1. Introducción

La construcción de la solución propuesta tiene como base el análisis de la esta a nivel de casos de uso. Aquí se representan, por cada caso de uso, los diagramas de clases de análisis y diseño, dando una vista de las clases que intervienen en el sistema (clase interfaz, clase controladora y clase entidad).

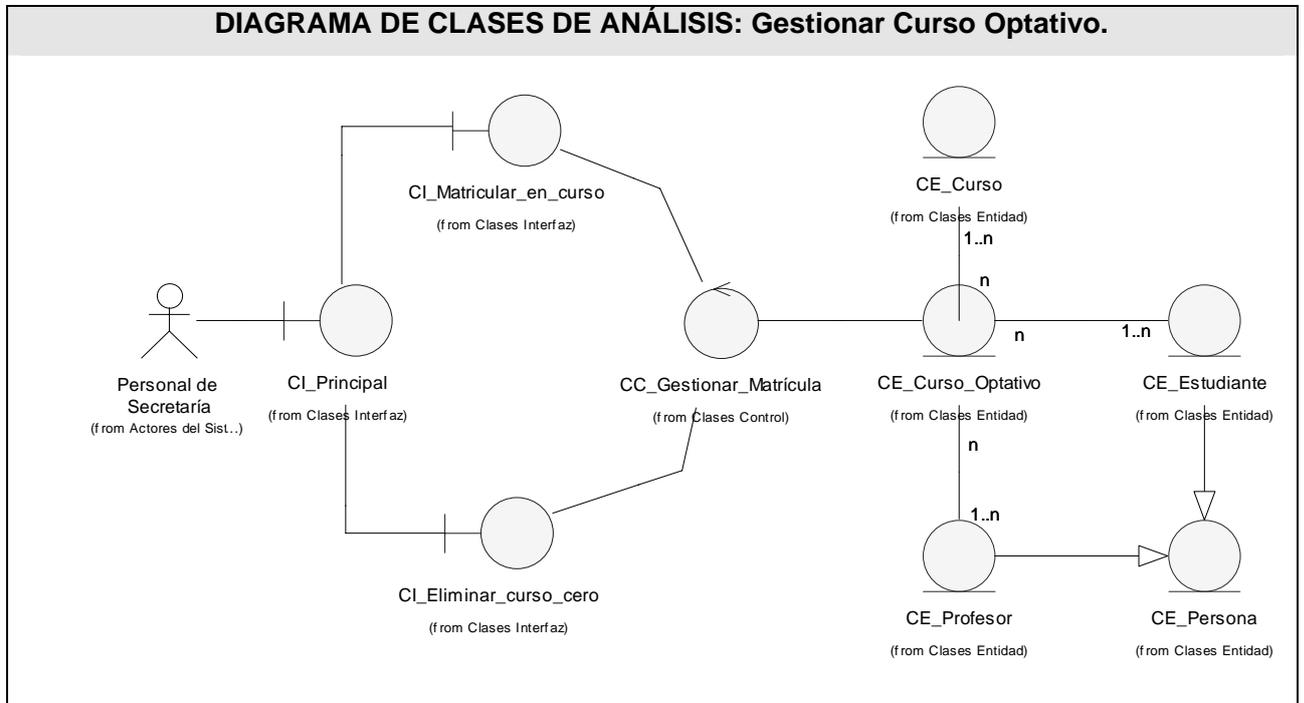
### 3.2. Análisis



**Fig 6** Diagrama de clases de análisis Autenticar Usuario.



**Fig 7** Diagrama de clases de análisis Gestionar Cursos.



**Fig 8** Diagrama de clases de análisis Gestionar Cursos.

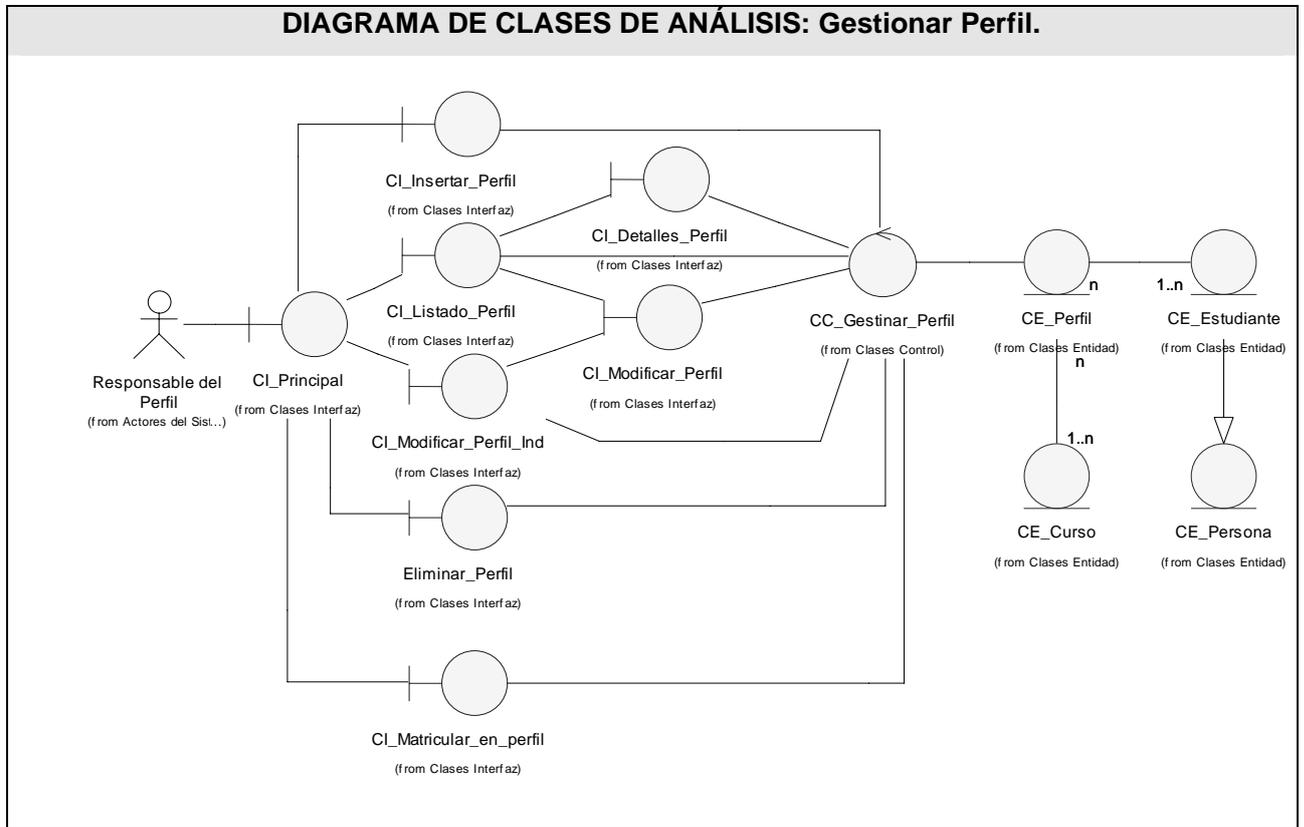
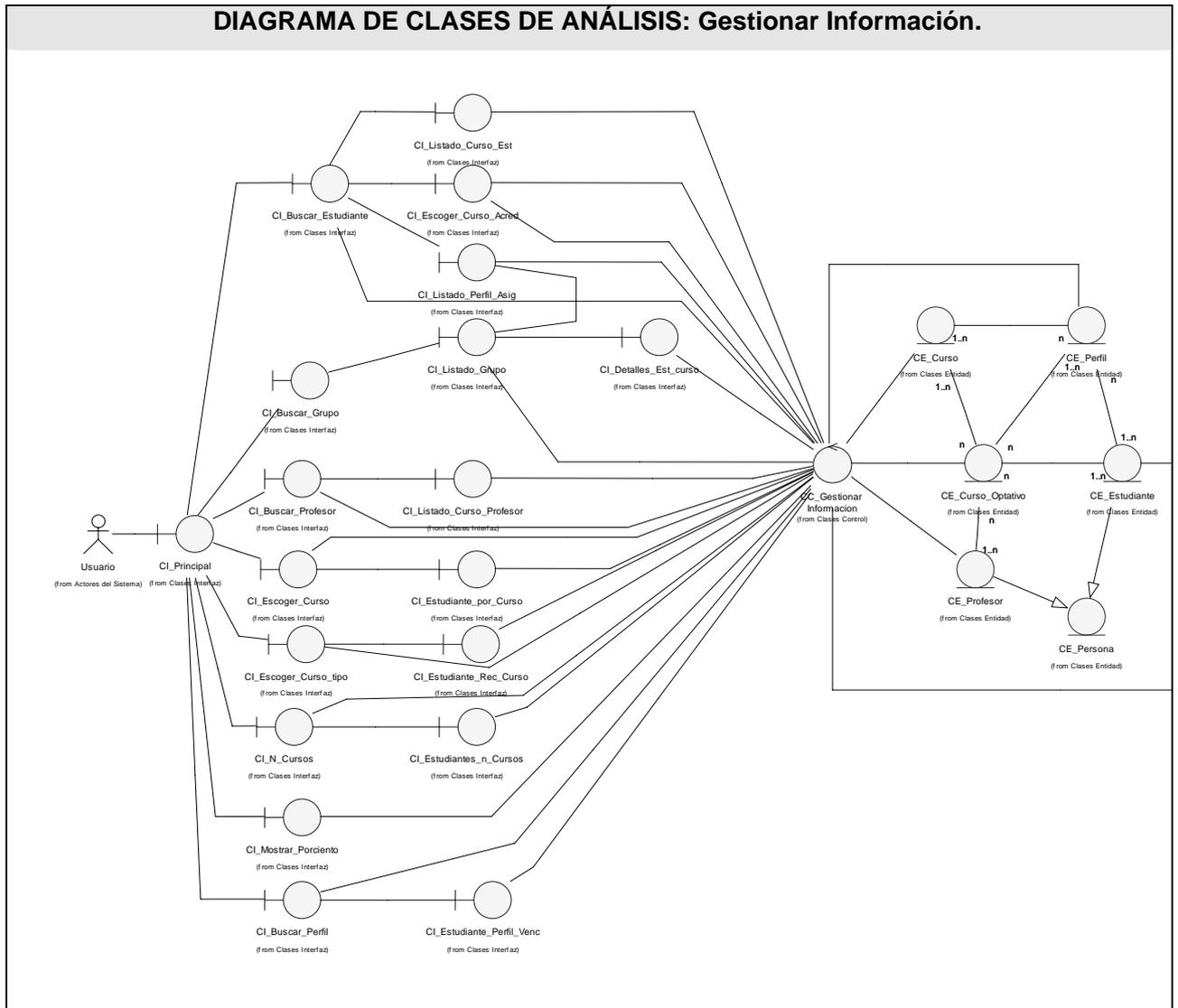


Fig 9 Diagrama de clases de análisis Gestionar Perfil.



**Fig 10** Diagrama de clases de análisis Gestionar Información.

### 3.3. Diseño

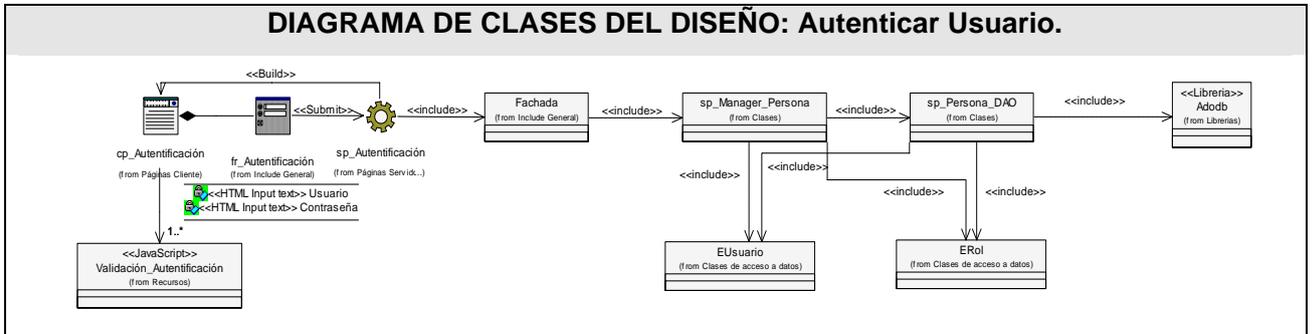


Fig 11 Diagrama de clases de diseño web Autenticar Usuario.

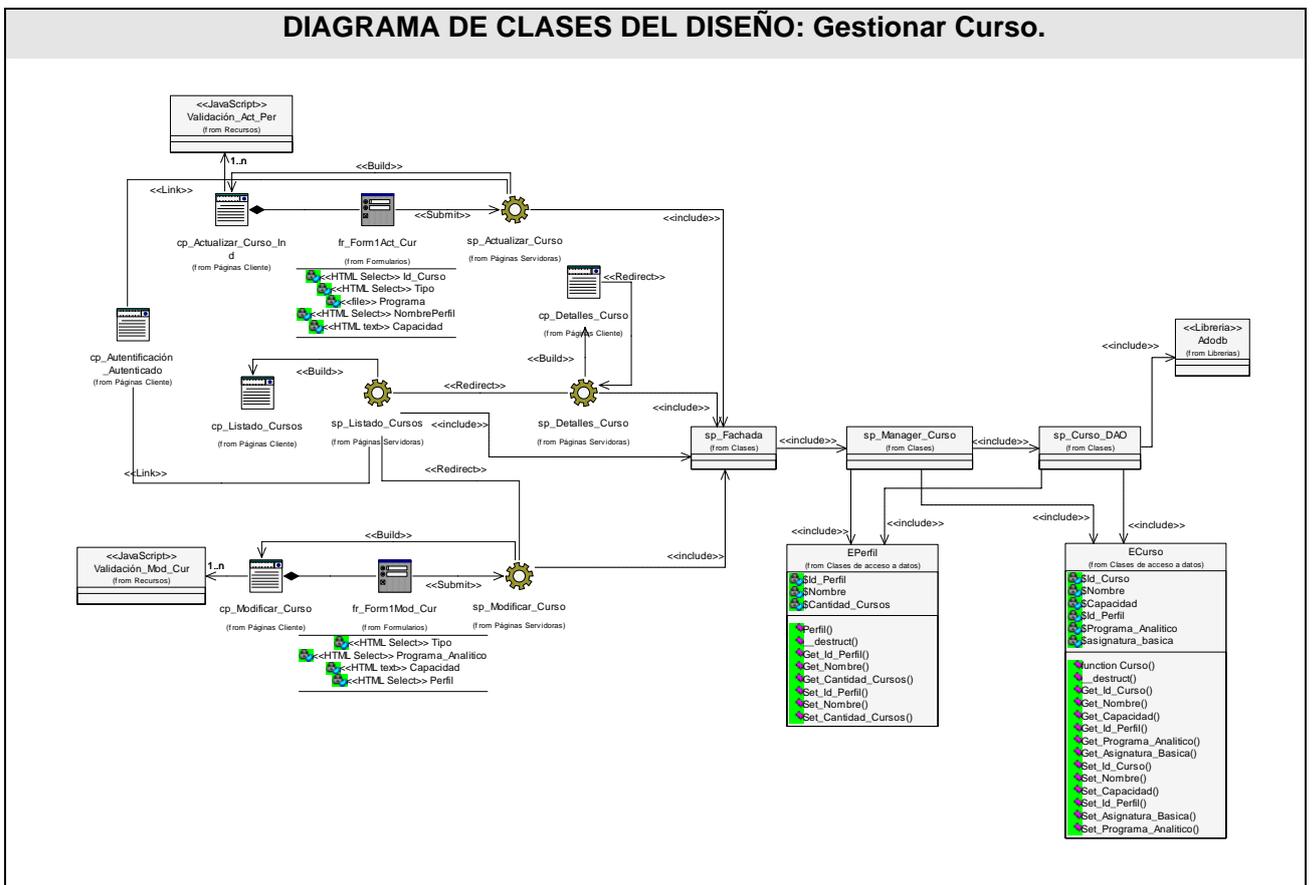


Fig 12 Diagrama de clases de diseño web Gestionar Cursos.

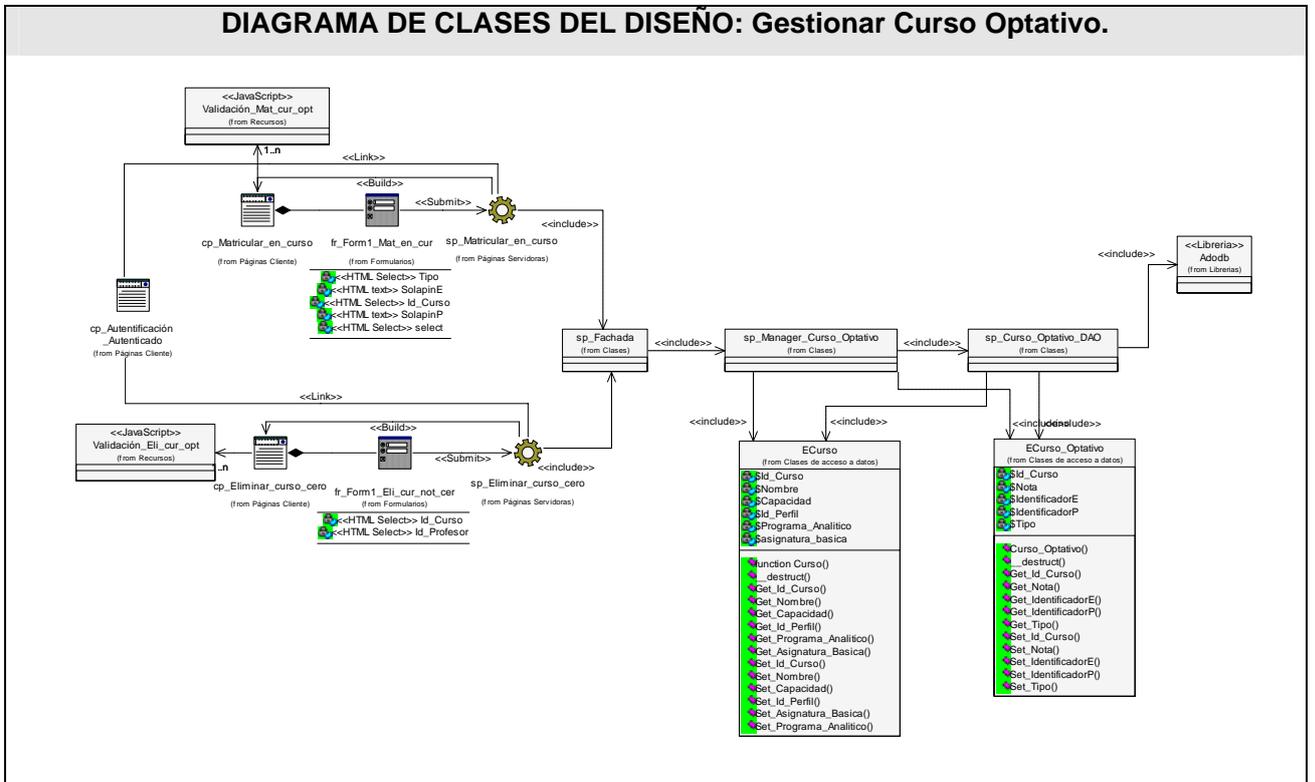


Fig 13 Diagrama de clases de diseño web Gestionar Curso Optativo.

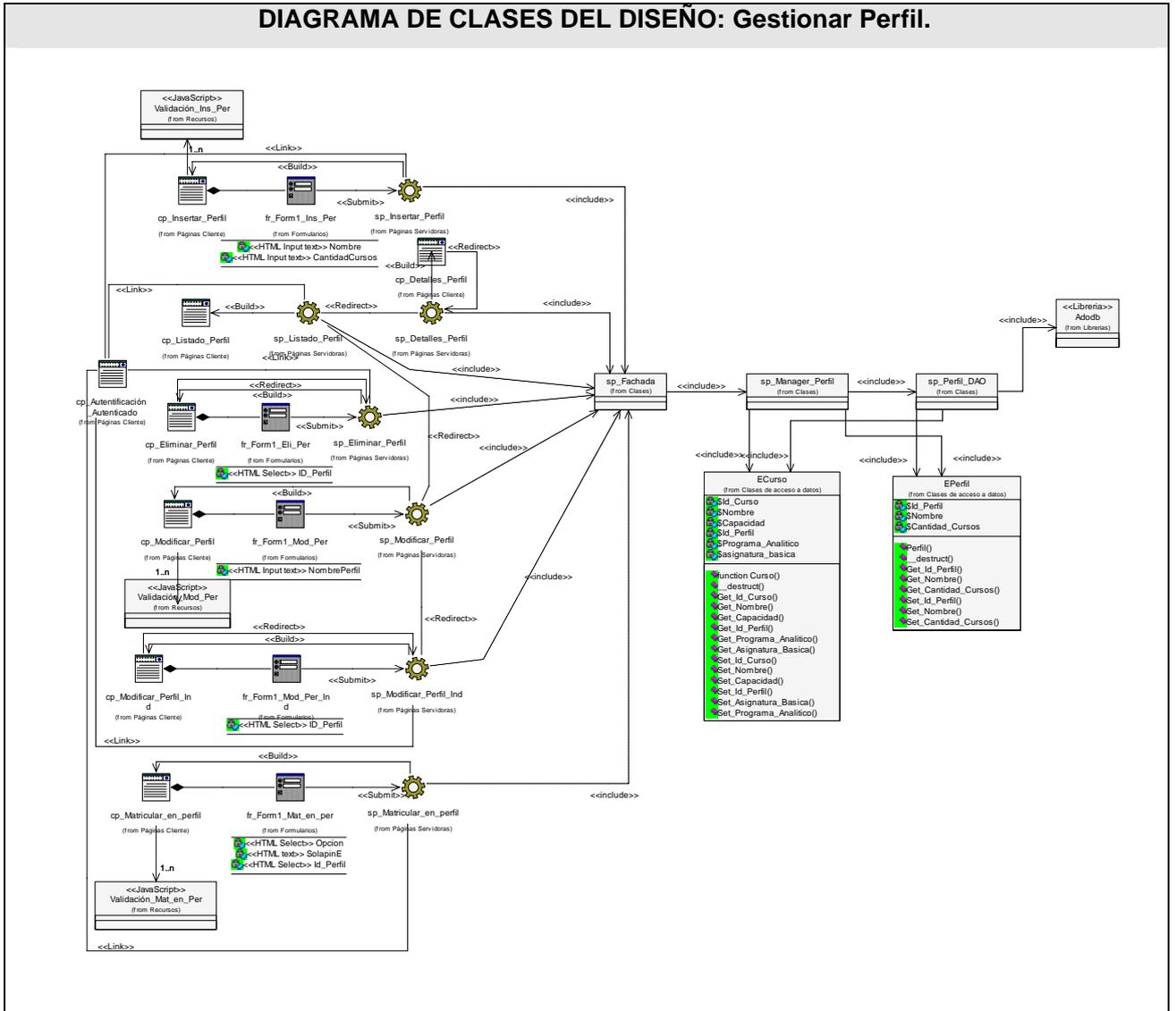


Fig 14 Diagrama de clases de diseño web Gestionar Perfil.

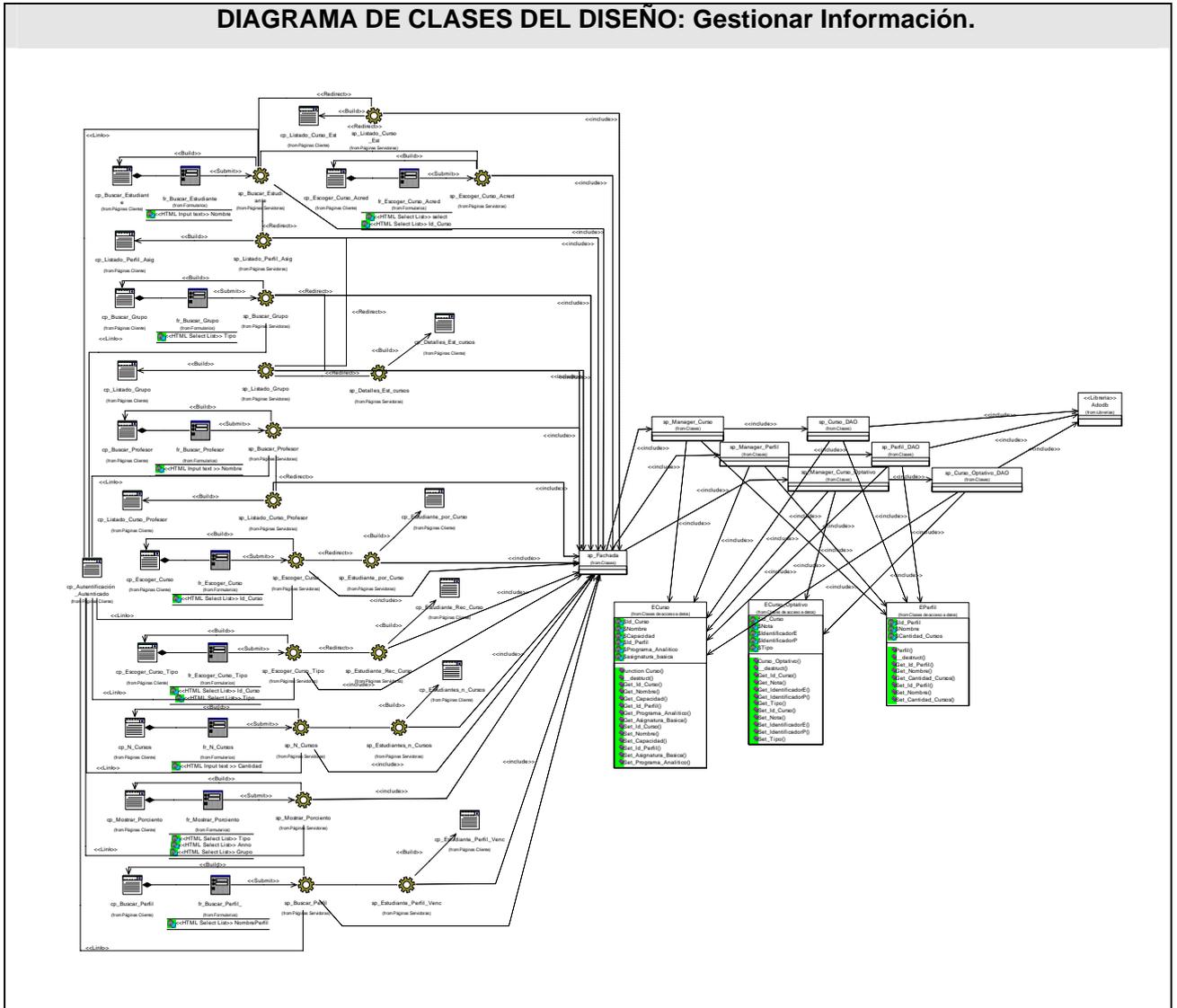


Fig 20 Diagrama de clases de diseño web Gestionar Información.

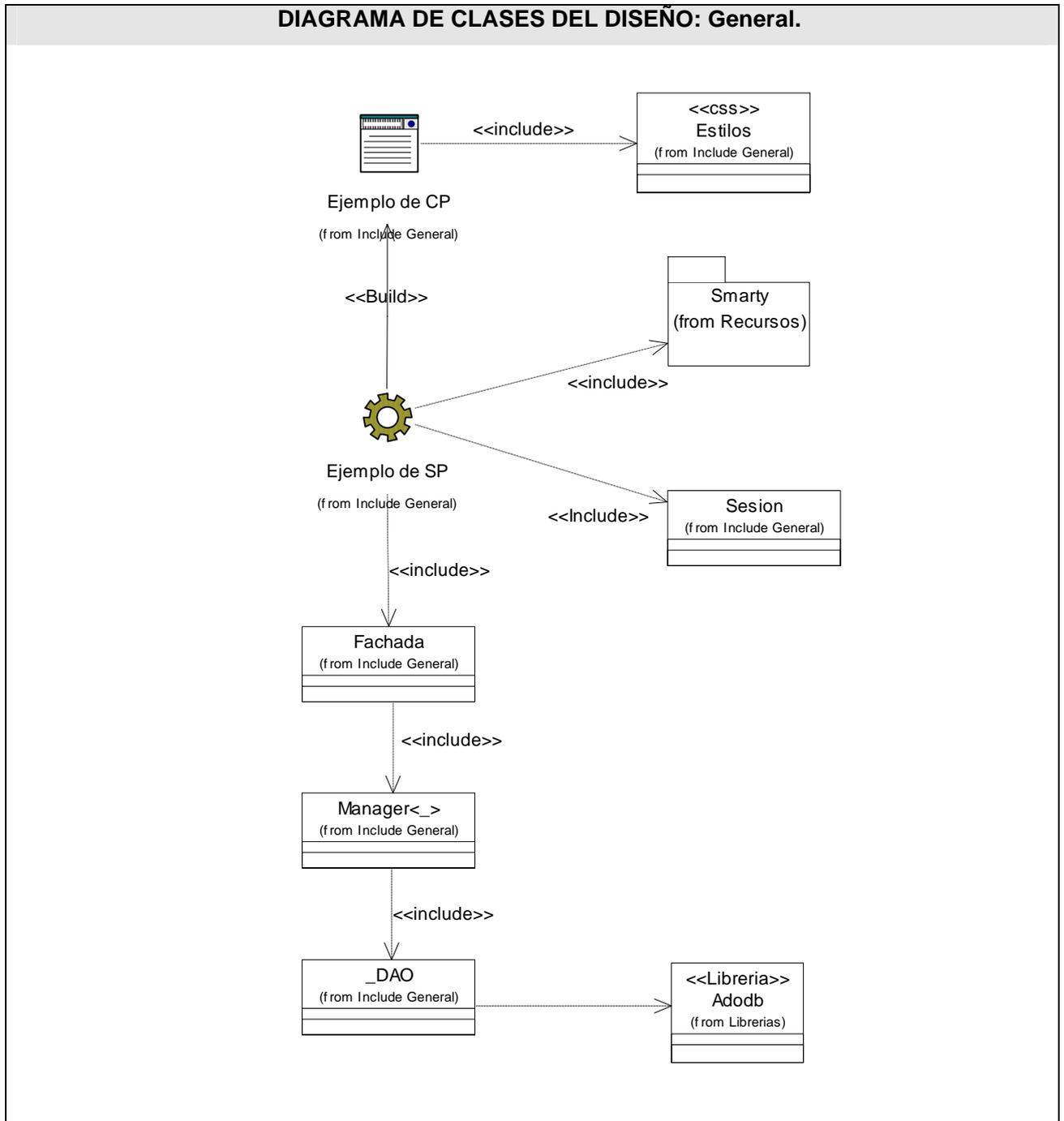


Fig 15 Diagrama General del funcionamiento de la programación y el paquete Incluye.

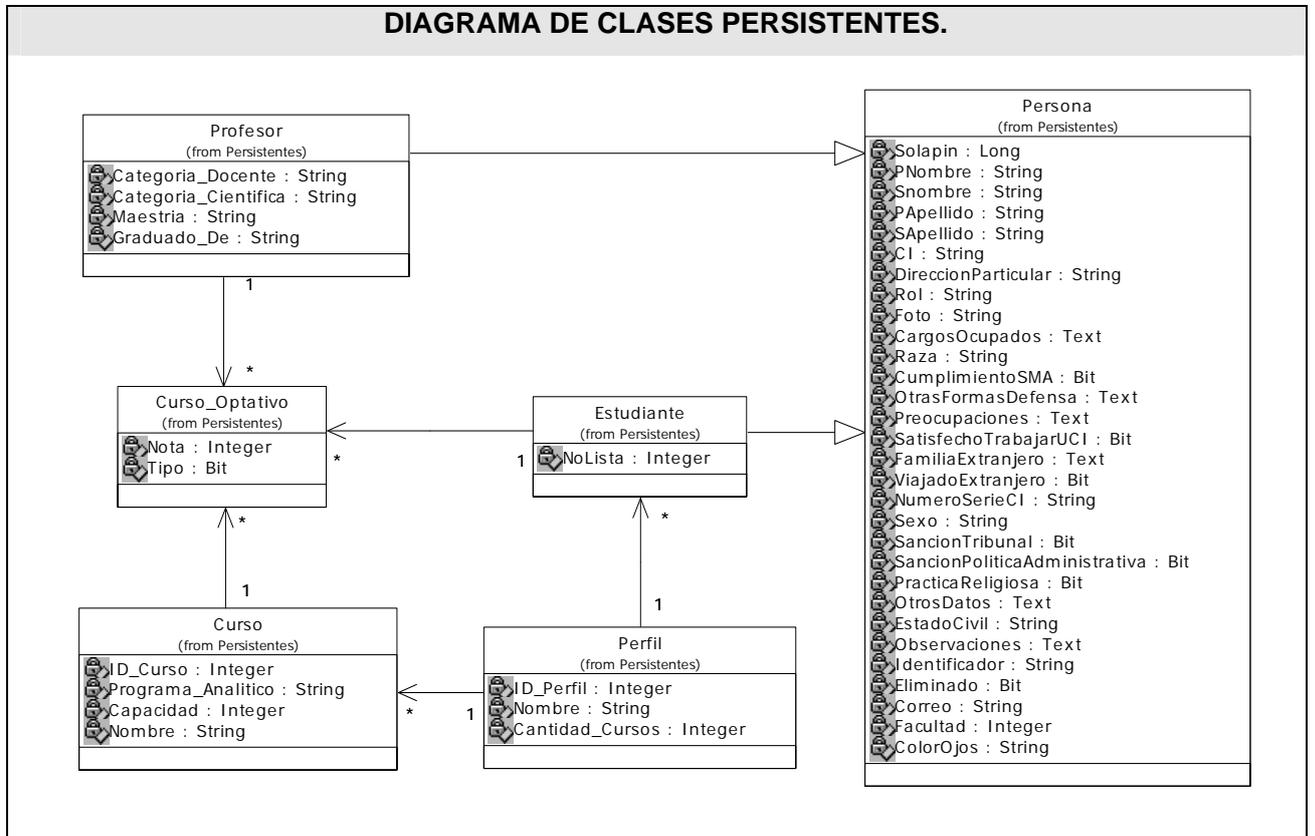


Fig 16 Diagrama de Clases Persistentes.

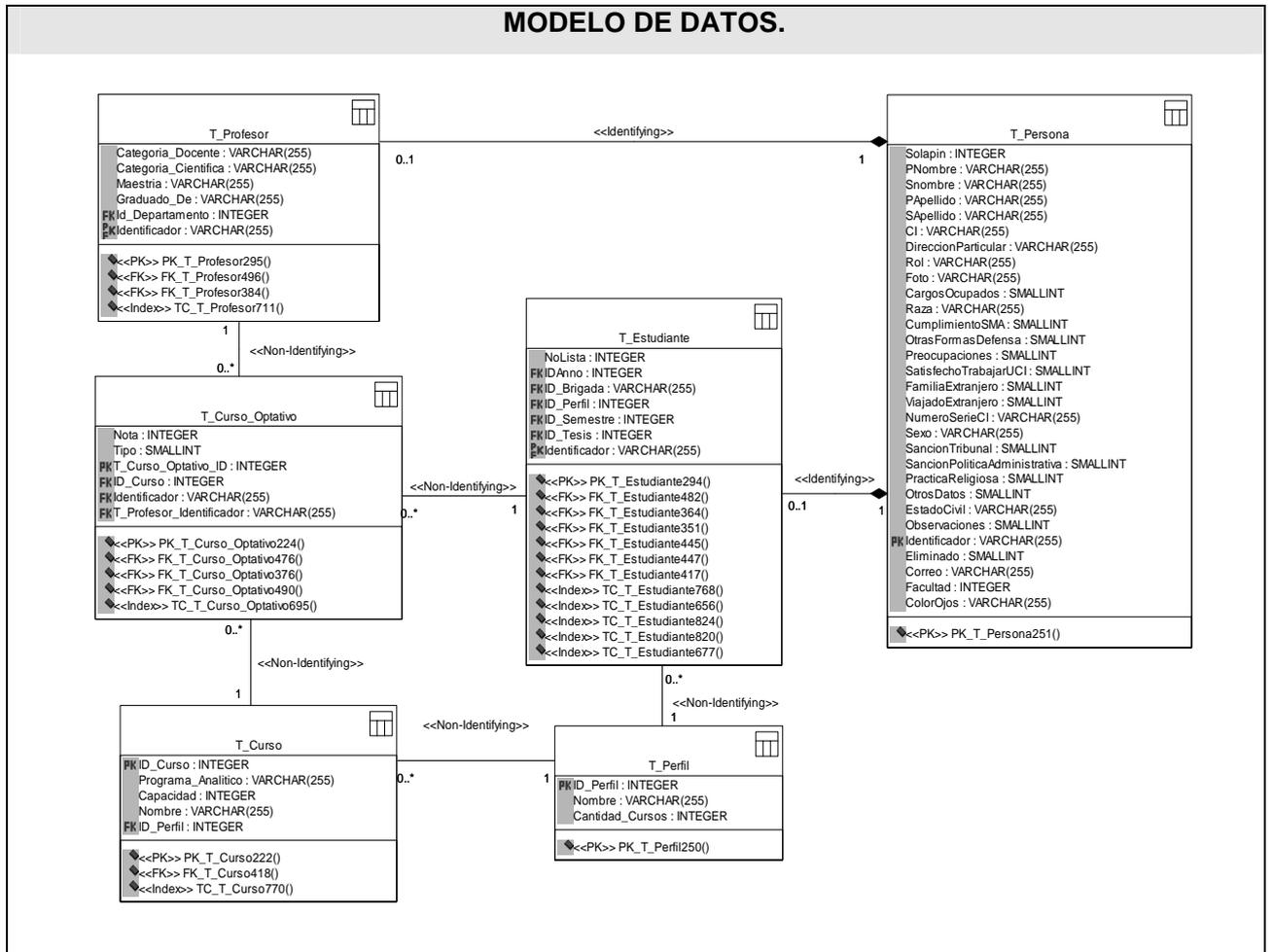


Fig 17 Modelo de Datos.

### 3.4. Descripción de las clases.

Table 12 Descripción de clases utilizadas en todos los CUS.

<b>Nombre: Autenticación</b>	
<b>Tipo de clase: Interfaz</b>	
<b>Atributo</b>	<b>Tipo</b>
Usuario	<<HTML Input >>
Contraseña	<<HTML Input >>
<b>Para cada responsabilidad:</b>	
Nombre:	
Descripción:	

<b>Nombre: Fachada</b>	
<b>Tipo de clase: Controladora</b>	
<b>Atributo</b>	<b>Tipo</b>
\$gestion_perfil	CManager_Perfil
\$gestion_curso	CManager_Curso
\$gestion_curso_optativo	CManager_Curso_Optativo
\$gestion_persona	CManager_Persona
\$gestion_profesor	CManager_Profesor
\$gestion_estudiante	CManager_Estudiante
<b>Para cada responsabilidad:</b>	

<b>Perfil</b> -----	
Nombre:	Insertar_Perfil(\$aNombre, \$aCantidad_Cursos) Insertar_Perfil_Grupo(\$ald_Brigada, \$ald_perfil) Listado_Curso_Optativo() Actualizar_Perfil(\$ald_Perfil, \$aNombre, \$aCantidad_Cursos) Eliminar_Perfil(\$aid_Perfil) Listado_Perfil() Cargar_Record_Perfil(\$aid_Perfil) Existe_Perfil(\$aNombre) Buscar_CursosOptativos(\$Id_Perfil) Buscar_AsignaturasBasicas(\$Id_Perfil) Cantidad_de_Perfiles() Listar_EstudiantesporPerfil(\$ald_Perfil) TotalEstudiantes_porPerfil(\$ald_Perfil) Estudiante_Perfil_Vencido(\$aldIdentificador)
<b>Curso</b> -----	
Nombre:	Actualizar_Curso(\$ald_Curso, \$aAsignatura_basica, \$aCapacidad, \$ald_Perfil, \$aPrograma_Analitico) Cargar_Record_Curso(\$aid_Curso) Listado_Curso() Cantidad_Cursos() Listado_Curso_Optativo_Cero()

<b>Curso Optativo</b> -----	
Nombre:	Eliminar_Curso_OptativoCero(\$ald_Curso, \$aldentificadorP) MostrarId_Brigadas() Mostrar_Estudiante_Acreditado_Curso(\$ald_Curso, \$aldentificadorE) BuscarProfesorCurso(\$ald_Curso) Mostrar_Estudiante_Apuntado_Curso(\$ald_Curso, \$aldentificadorE) CapacidadCursoOptativo(\$ald_Curso, \$aldentificadorP) Insertar_Curso_Optativo(\$ald_Curso, \$aldentificadorE, \$aldentificadorP, \$aNota, \$aTipo) Listar_CursosporEstudiante(\$aldentificador) Listar_EstudiantesporCurso(\$Id_Curso) Listar_EstudiantesconNCursos(\$CantidadCursos) Cantidad_EstudiantesconNCursos(\$CantidadCursos) TotalEstudiantes_porCurso(\$Id_Curso) Porciento_por_Año(\$aAño) Porciento_por_Grupo(\$aGrupo) Lista_EstudiantesRecibiendoCursoOptativo(\$ald_Curso, \$aTipo) Lista_AsignaturasBasicasTermEstudiantes(\$aldentificadorE)
<b>Persona</b> -----	
Nombre:	Listado_Persona() Cargar_Record_Persona(\$aldentificador) Existe_Persona(\$aSolapin) Es_Docente(\$aldentificador)
<b>Estudiante</b> -----	
Nombre:	Listado_Estudiante() Insertar_Perfil_Estudiante(\$aldentificador, \$aid_Perfil) Cargar_Record_Estudiante(\$aldentificador) Existe_Estudiante(\$aldentificador)

	Buscar_Perfil_Estudiante(\$aldentificador, \$aid_Perfil) FCargar_Record_EstudiantesGrupo(\$ald_Brigada) Buscar_Estudiante(\$aNombre) Buscar_Estudiante_Completo(\$aPNombre, \$aSNombre, \$aPApellido, \$aSApellido) Cantidad_CursosEst(\$aldentificador) PerfilEstudiante(\$aldentificador)
<b>Profesor</b> -----	
Nombre:	Listado_Profesor() Cargar_Record_Profesor(\$aSolapin) Existe_Profesor(\$aSolapin) Buscar_Profesor(\$aNombre) Cantidad_CursosPro(\$aldentificador) Listar_CursosporProfesor(\$aldentificador)
Descripción: General	Estas funciones son las encargadas de recibir los datos y distribuirlos a las clases manager correspondientes de acuerdo a la acción solicitada desde las páginas servidoras superiores.

**Table 13** Descripción de clases utilizadas en el CUS Gestionar Curso Optativo.

<b>Nombre: Matricular_en_curso</b>	
<b>Tipo de clase: Interfaz</b>	
<b>Atributo</b>	<b>Tipo</b>
Tipo	<<HTML select>>
SolapinE	<<HTML input>>
Id_Curso	<<HTML select>>
SolapinP	<<HTML input>>
Modo	<<HTML select>>

<b>Para cada responsabilidad:</b>	
Nombre:	Verifica()
Descripción:	Validar los datos.
Nombre:	Cambio()
Descripción:	Mostrar y ocultar campos en correspondencia con lo seleccionado por el usuario.

<b>Nombre: Eliminar_Curso_Cero</b>	
<b>Tipo de clase: Interfaz</b>	
<b>Atributo</b>	<b>Tipo</b>
Id_Curso	<<HTML select>>
Id_Profesor	<<HTML select>>
<b>Para cada responsabilidad:</b>	
Nombre:	Verifica()
Descripción:	Validar los datos
Nombre:	Buscar()
Descripción:	Enviar los datos

<b>Nombre: CManager_Curso_Optativo</b>	
<b>Tipo de clase: Controladora</b>	
<b>Atributo</b>	<b>Tipo</b>

<b>Para cada responsabilidad:</b>	
<b>Nombre:</b>	Listado_Curso_Optativo() Eliminar_Curso_OptativoCero(\$ald_Curso, \$aldentificadorP) MostrarId_Brigadas() Mostrar_Estudiante_Acreditado_Curso(\$ald_Curso, \$aldentificadorE) BuscarProfesorCurso(\$ald_Curso) Mostrar_Estudiante_Apuntado_Curso(\$ald_Curso, \$aldentificadorE) CapacidadCursoOptativo(\$ald_Curso, \$aldentificadorP) Insertar_Curso_Optativo(\$ald_Curso, \$aldentificadorE, \$aldentificadorP, \$aNota, \$aTipo) Listar_CursosporEstudiante(\$aldentificadorE) Listar_EstudiantesporCurso(\$Id_Curso) Listar_EstudiantesconNCursos(\$CantidadCursos) Cantidad_EstudiantesconNCursos(\$CantidadCursos) TotalEstudiantes_porCurso(\$Id_Curso) Porcentaje_por_Año(\$aAño) Porcentaje_por_Grupo(\$aGrupo) Lista_EstudiantesRecibiendoCursoOptativo(\$ald_Curso, \$aTipo) Lista_AsignaturasBasicasTermEstudiantes(\$aldentificadorE)
<b>Descripción:</b>	Crean los objetos de la clase entidad Curso_Optativo a partir de los datos recibidos de la clase fachada y llaman a las funciones de acceso a datos correspondientes pasándole esos objetos para que estas realicen el trabajo en la base de datos.

<b>Nombre: Curso_Optativo_DAO</b>	
<b>Tipo de clase: Controladora</b>	
<b>Atributo</b>	<b>Tipo</b>

<b>Para cada responsabilidad:</b>	
<b>Nombre:</b>	Listado_Curso_Optativo() Eliminar_Curso_OptativoCero(Curso_Optativo \$Curso_Optativo) MostrarId_Brigadas() Mostrar_Estudiante_Acreditado_Curso(Curso_Optativo \$Curso_Optativo) BuscarProfesorCurso(Curso_Optativo \$Curso_Optativo) Mostrar_Estudiante_Apuntado_Curso(Curso_Optativo \$Curso_Optativo) CapacidadCursoOptativo(Curso_Optativo \$Curso_Optativo) Insertar_Curso_Optativo(Curso_Optativo \$Curso_Optativo) Listar_CursosporEstudiante(Curso_Optativo \$Curso_Optativo) Listar_EstudiantesporCurso(Curso_Optativo \$Curso_Optativo) Listar_EstudiantesconNCursos(\$CantidadCursos) Cantidad_EstudiantesconNCursos(\$CantidadCursos) TotalEstudiantes_porCurso(Curso_Optativo \$Curso_Optativo) Porcentaje_por_Año(\$aAño) Porcentaje_por_Grupo(\$aGrupo) Lista_EstudiantesRecibiendoCursoOptativo(Curso_Optativo \$Curso_Optativo) Lista_AsignaturasBasicasTermEstudiantes(Curso_Optativo \$Curso_Optativo)
<b>Descripción:</b>	A partir de las peticiones recibidas de la clase manager correspondiente mandan a ejecutar acciones sobre la base de datos.

<b>Nombre: Curso_Optativo</b>	
<b>Tipo de clase: Entidad</b>	
<b>Atributo</b>	<b>Tipo</b>
\$Id_Curso	Se definen en tiempo de ejecución de acuerdo a lo recibido teniendo en cuenta que estos deben ser compatibles con los tipos que están almacenados en la base de datos.
\$Nota	
\$IdentificadorE	
\$IdentificadorP	

\$Tipo	
<b>Para cada responsabilidad:</b>	
Nombre:	Curso_Optativo(\$ald_Curso = null, \$aNota = null, \$aldentificadorE = null, \$aldentificadorP = null, \$aTipo = null)
Descripción:	Crear el objeto con los parámetros pasados, en caso de no ser pasados entran con <b>null</b> .
Nombre:	__destruct()
Descripción:	Destruir el objeto.
Nombre:	Get_Id_Curso() Get_Nota() Get_IdentificadorE() Get_IdentificadorP() Get_Tipo()
Descripción:	Devolver el valor de los atributos.
Nombre:	Set_Id_Curso(\$ald_Curso) Set_Nota(\$aNota) Set_IdentificadorE(\$aldentificadorE) Set_IdentificadorP(\$aldentificadorP) Set_Tipo(\$aTipo)
Descripción:	Actualizar el valor de los atributos con los datos que son pasados.

El resto de las descripciones están en los **Anexos referentes a la descripción de las clases del diseño**.

### 3.5. Conclusiones

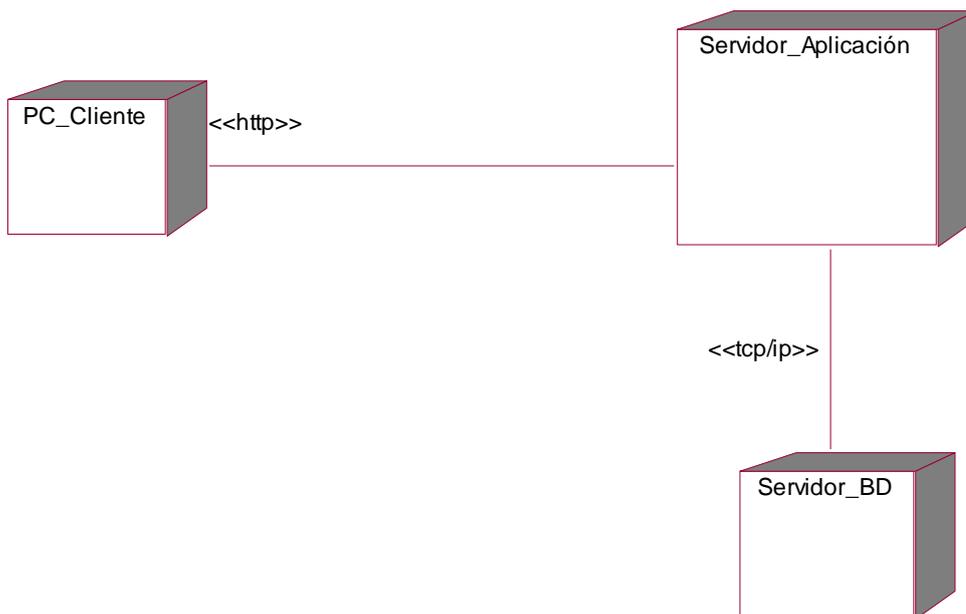
Han sido modelados algunos artefactos del flujo de trabajo de análisis y diseño que serán implementados, lo cual debe cumplir con todo lo establecido para que tenga la calidad que se requiere y sea amigable con el usuario.

## CAPÍTULO 4 IMPLEMENTACIÓN Y PRUEBA

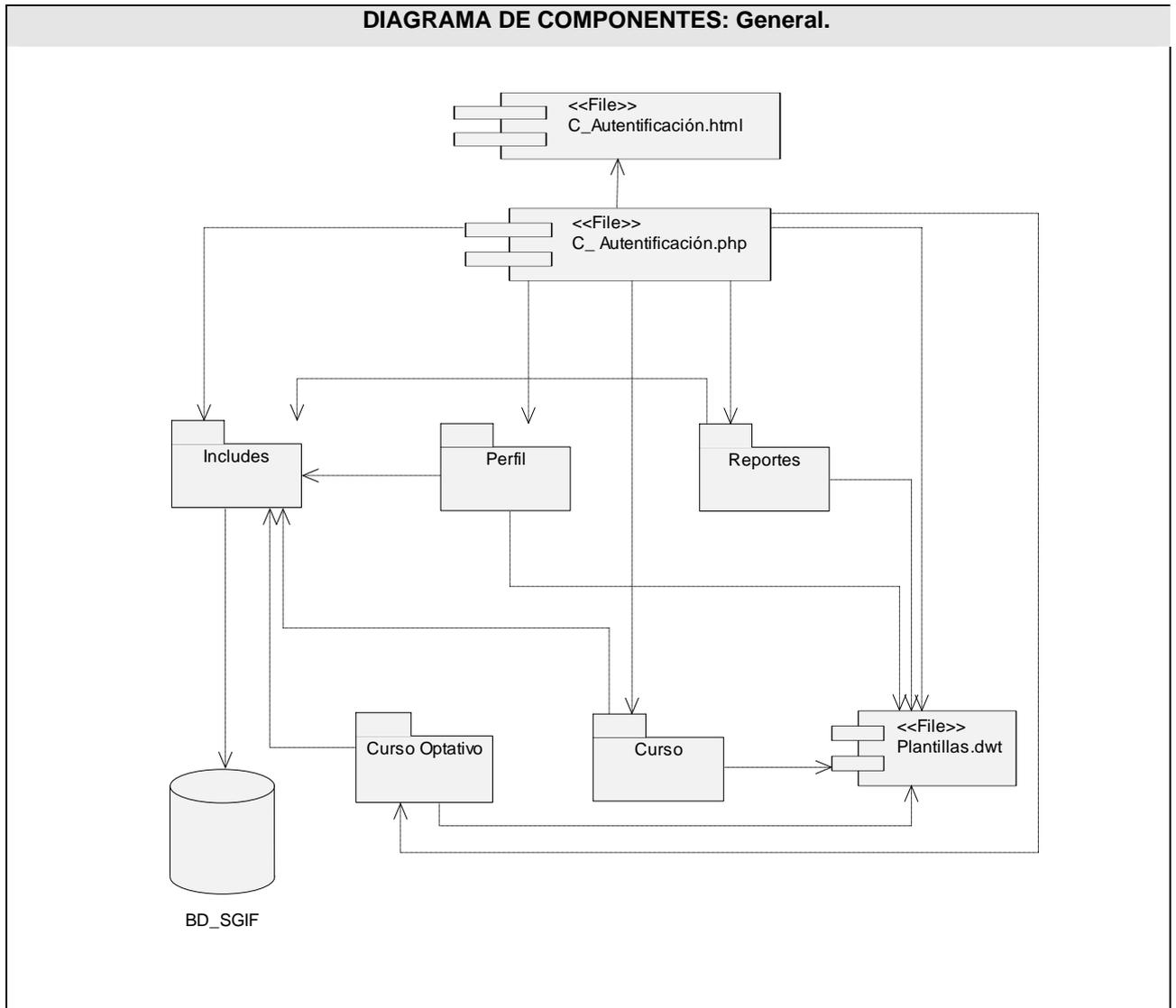
### 4.1. Introducción

En este capítulo se muestran los artefactos que representan la organización física que soportará la instalación del sistema, así como los componentes software que se ejecutarán en cada uno de estos dispositivos conformando de forma general el producto, respetando la arquitectura que se utilizó en la solución del problema. Se describen las pruebas de caja negra hechas al software.

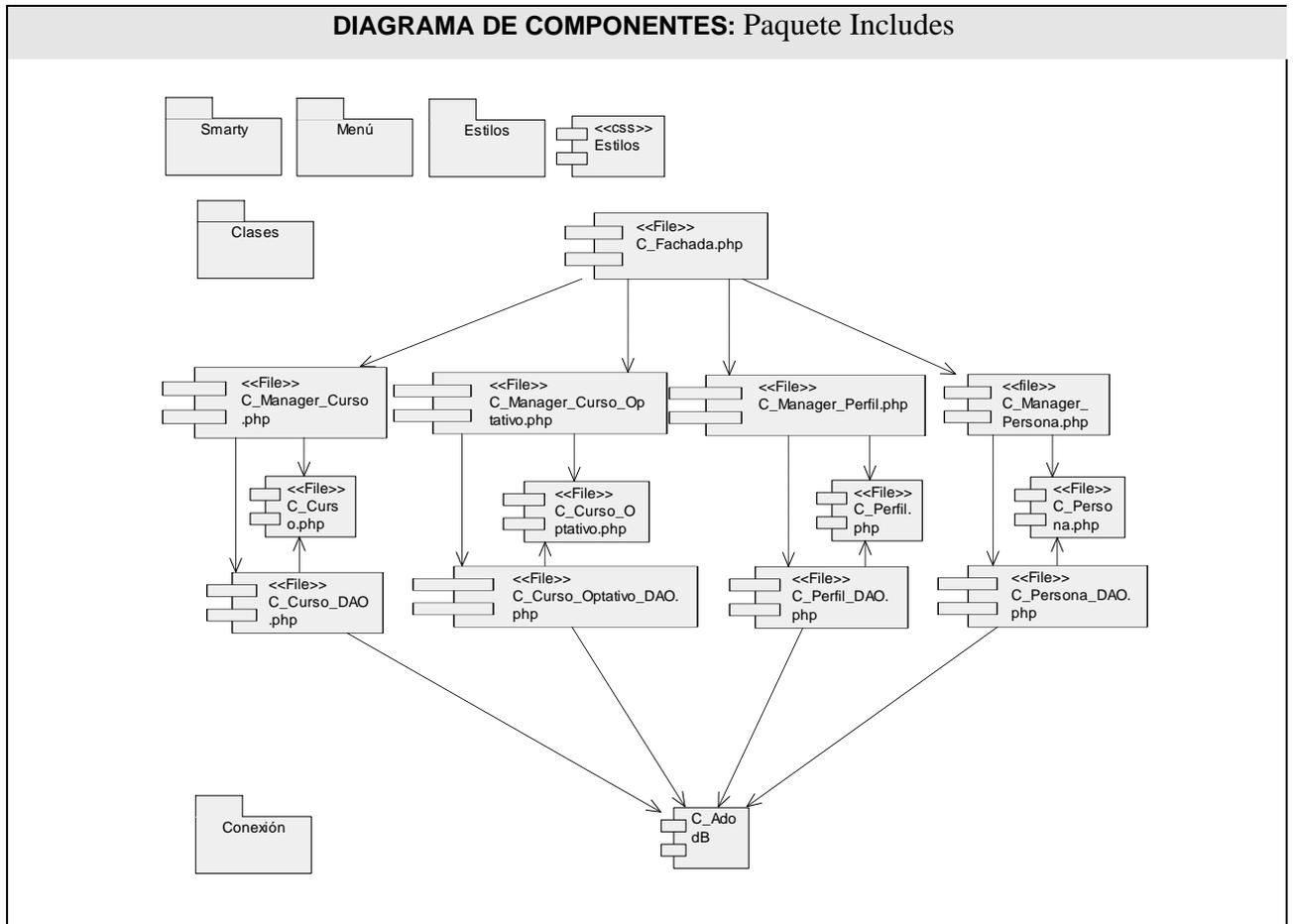
### 4.2. Implementación



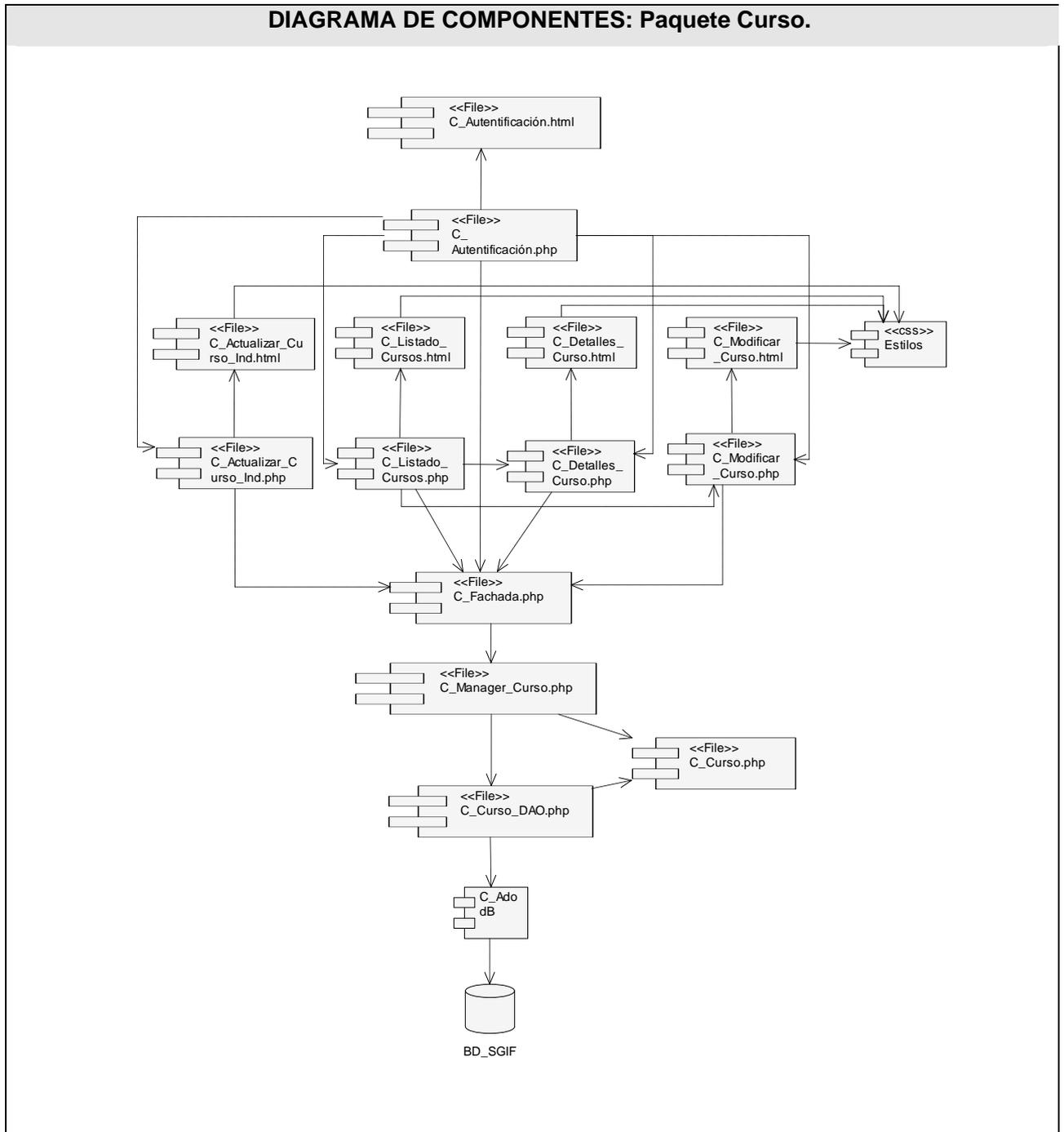
**Fig 18** Diagrama de despliegue.



**Fig 19** Diagrama de componentes general.



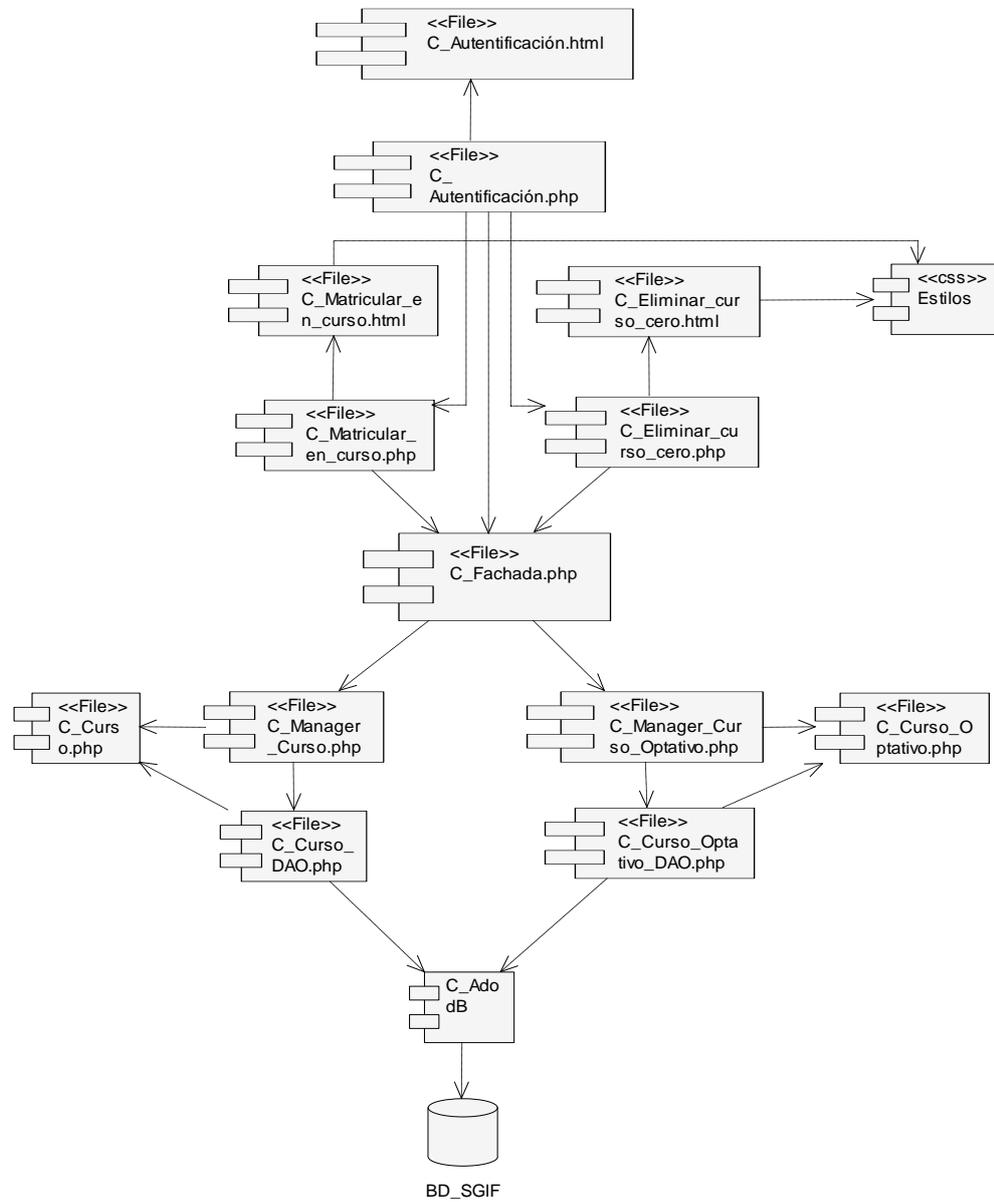
**Fig 20** Diagrama de componentes del Paquete Includes.



**Fig 21** Diagrama de componentes del Paquete Curso.

DIAGRAMA DE COMPONENTES: Paquete Curso Optativo.

**Fig 22** Diagrama de componentes del Paquete Curso Optativo.



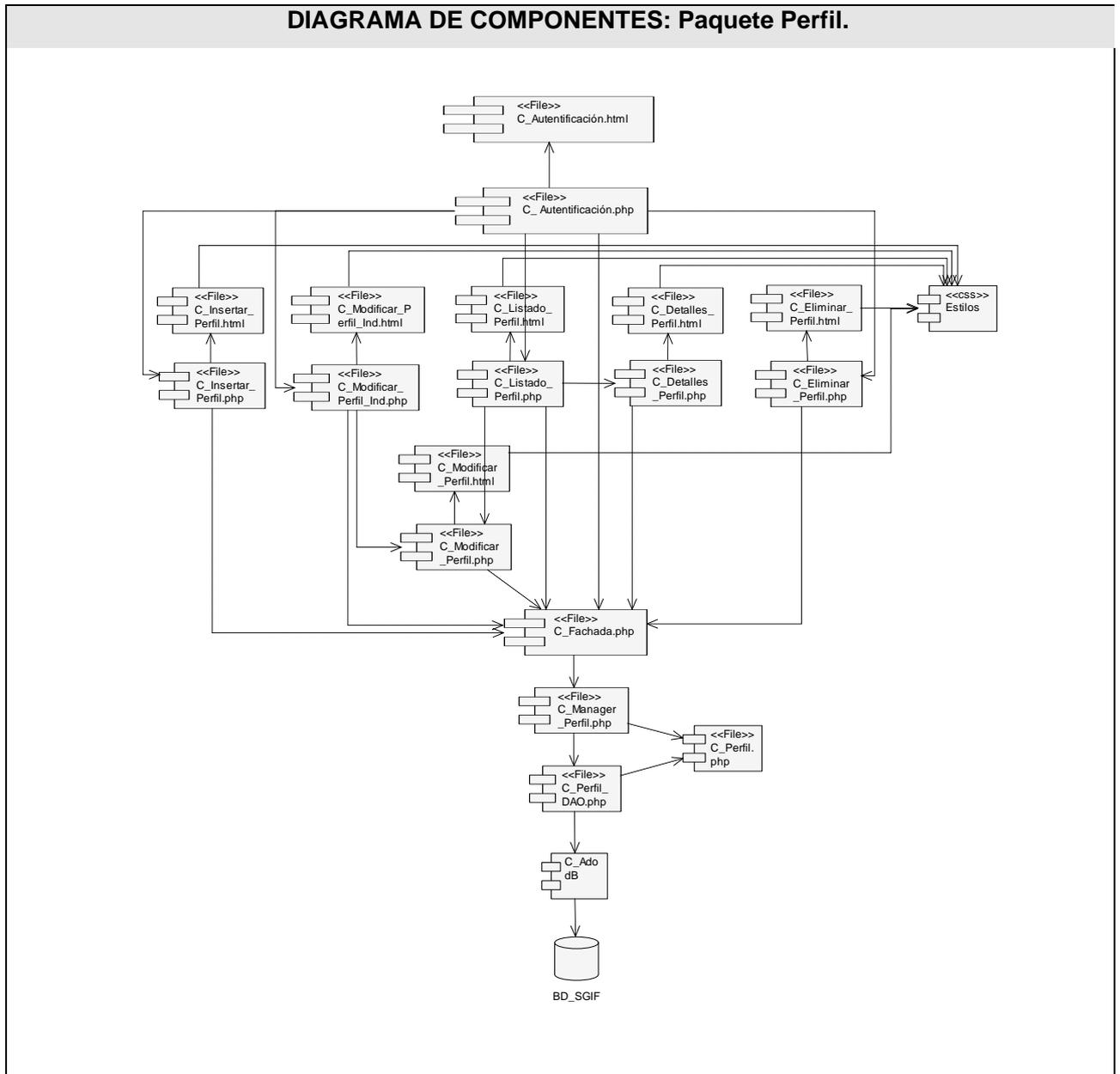
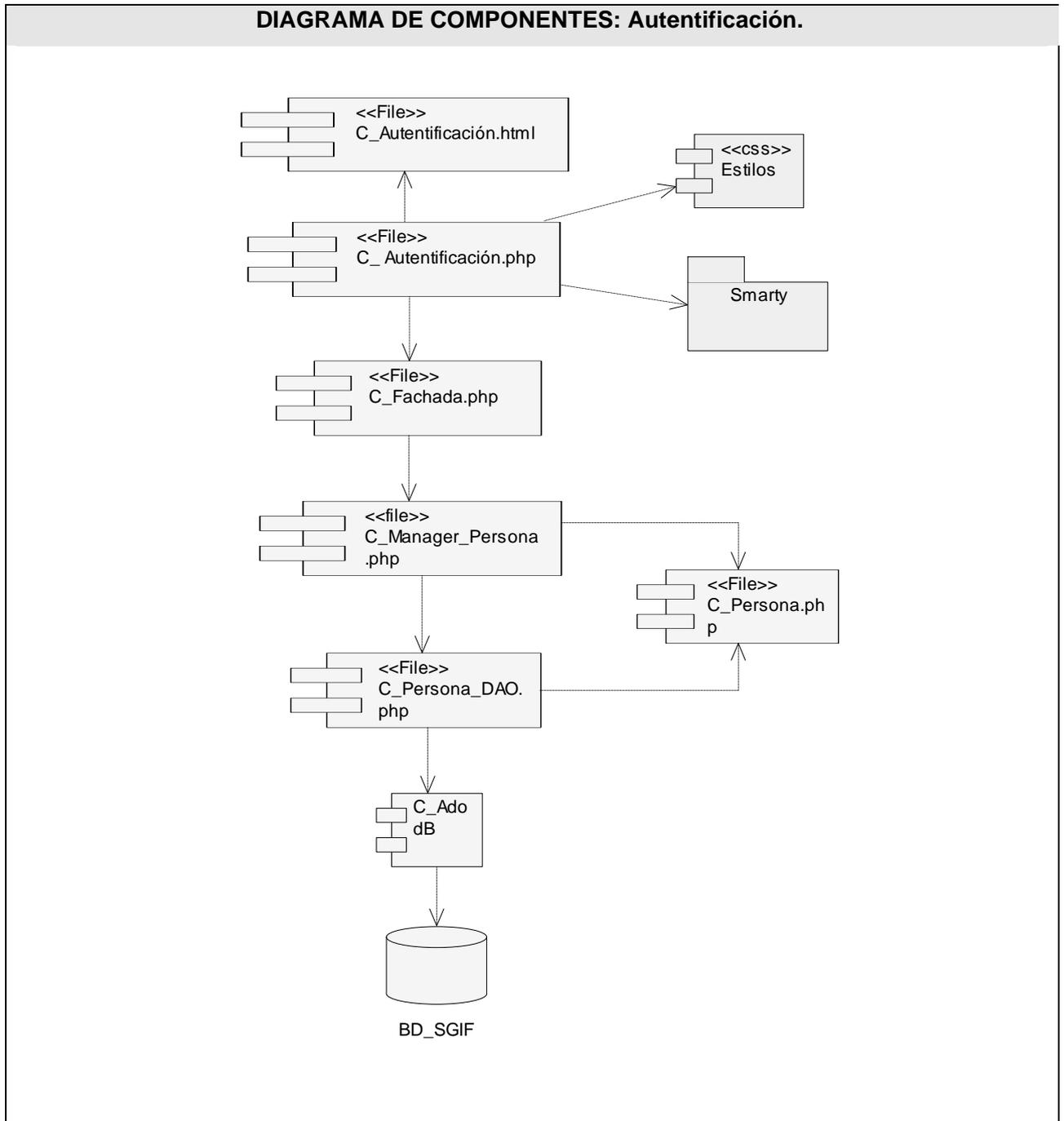


Fig 23 Diagrama de componentes del Paquete Perfil.





**Fig 24** Diagrama de componentes Autenticación.

### 4.3. Modelo de prueba

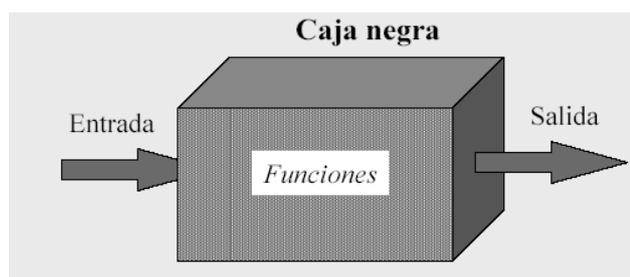
#### ¿Qué es una prueba?

La prueba del software es un conjunto de actividades que se lleva a cabo sistemáticamente, que puede planificarse por adelantado y ejecutar una vez construido el código para la revisión final de las especificaciones, del diseño y de la codificación del software.

Cualquier producto de ingeniería puede ser probado mediante una de estas técnicas:

1. Conociendo la funcionalidad específica para la cual fue diseñado el producto, se pueden llevar a cabo pruebas que demuestren que cada función es completamente operativa. **Pruebas de caja negra.**
2. Conociendo el funcionamiento del producto se pueden desarrollar pruebas que aseguren que “todas las piezas encajen”, o sea, que la operación interna se ajusta a las especificaciones y que todos los componentes internos se han comprobado de forma adecuada. **Pruebas de caja blanca.**

Para el sistema desarrollado se le realizaron pruebas de caja negra, la cual se centra fundamentalmente en los *requerimientos funcionales* del software.



**Fig 25** Representación de la Prueba de Caja negra.

Estas pruebas permiten obtener un conjunto de condiciones de entrada que ejerciten completamente todos los requisitos funcionales de un programa. En ellas se ignora la estructura de control, concentrándose en los requisitos funcionales del sistema y ejercitándolos.

Casos de prueba

**Table 14** Caso de Prueba del CUS Gestionar Curso Optativo: Matricular en un Curso.

Nombre del caso de uso:	Gestionar Curso Optativo
Nombre del caso de prueba	Matricular en un Curso
Entradas	Estructura:Estudiante Solapín del Estudiante::54839 Nombre del Curso:css Solapín del Profesor del Curso:2001 Tipo de Curso: Normal
Resultados esperados	Proceso Satisfactorio.
Condiciones	Que el estudiante no lo tenga cursado o matriculado
Resultados de la prueba	Proceso Satisfactorio.

**Table 15** Caso de Prueba del CUS Gestionar Curso Optativo: Matricular en un Curso.

Nombre del caso de uso:	Gestionar Curso Optativo
Nombre del caso de prueba	Matricular en un Curso

Entradas	Estructura:Estudiante Solapín del Estudiante::54839 Nombre del Curso:css Solapín del Profesor del Curso:2001 Tipo de Curso: Normal
Resultados esperados	Error: El Estudiante ya está apuntado o tiene acreditado ese Curso
Condiciones	Que el estudiante tenga cursado o matriculado un curso.
Resultados de la prueba	Error: El Estudiante ya está apuntado o tiene acreditado ese Curso

**Table 16** Caso de Prueba del CUS Gestionar Curso Optativo: Matricular en un Curso.

<b>Nombre del caso de uso:</b>	<b>Gestionar Curso Optativo</b>
Nombre del caso de prueba	Matricular en un Curso
Entradas	Estructura:Estudiante Solapín del Estudiante::54839 Nombre del Curso:css Solapín del Profesor del Curso:2001 Tipo de Curso: Normal
Resultados esperados	Error: Ese Solapín de Estudiante no esta en la Base de datos.
Condiciones	Que el solapin del estudiante no este en la base

	de datos
Resultados de la prueba	Error: Ese Solapín de Estudiante no esta en la Base de datos.

**Table 17** Caso de Prueba del CUS Gestionar Curso Optativo: Matricular en un Curso.

<b>Nombre del caso de uso:</b>	<b>Gestionar Curso Optativo</b>
Nombre del caso de prueba	Matricular en un Curso
Entradas	Estructura:8501 Nombre del Curso:css Solapín del Profesor del Curso:2001 Tipo de Curso: Normal
Resultados esperados	Se Matricularon 5 Estudiantes, había 0 apuntados de 5 que hay en el Grupo
Condiciones	Que en el grupo solo haya 5 estudiante sy que ninguno este apuntado o lo tenga acreditado
Resultados de la prueba	Se Matricularon 5 Estudiantes, había 0 apuntados de 5 que hay en el Grupo

**Table 18** Caso de Prueba del CUS Gestionar Curso Optativo: Eliminar Curso con nota 0.

<b>Nombre del caso de uso:</b>	<b>Gestionar Curso Optativo</b>
Nombre del caso de prueba	Eliminar Curso con nota 0
Entradas	Nombre del Curso: htlm Nombre del Profesor: Alex Ortega

Resultados esperados	Proceso Satisfactorio.
Condiciones	El profesor alex ortega debe de tener estudiantes pasando ese curso.
Resultados de la prueba	Proceso Satisfactorio.

El resto de los casos de prueba están en los **Anexos referentes a los casos de prueba.**

#### **4.4. Conclusiones**

Como resultado en este capítulo se han creado los artefactos pertenecientes a esta etapa del proyecto ej: diagramas de despliegue, digrama de componentes. Estos artefactos representan cómo quedara el despliegue del software en la etapa de instalación.

## CONCLUSIONES

Toda construcción de un software informático y la documentación pertinente conlleva a resultados. Es por ello, que guiados por el proceso de desarrollo de software RUP, se realizó una aplicación web para la gestión de la información relacionada con el perfil y los cursos optativos de la Facultad #8, pudiéndose decir que la explotación de este producto, pretende mejorar la gestión y control de esta.

Se concluye que:

- ✓ Se procesó toda la información inherente a la gestión de los cursos optativos y el perfil, haciendo énfasis en sus principales reportes, personal implicado y procesos que ocurren en la entidad.
- ✓ Se realizó el análisis, diseño e implementación del subsistema para la gestión de la información anterior, cumpliéndose los requerimientos definidos por el cliente.
- ✓ Se implementó el proceso de matrícula para que un estudiante pueda recibir cualquier curso optativo y/o un perfil.
- ✓ Se generaron en formato Excel los reportes más significativos, con el objetivo de manipular la información y poderla imprimir en algún momento.
- ✓ Se creó un documento donde se recogió todo el proceso investigativo del desarrollo del sistema informático, con la modelación de los artefactos obtenidos durante su ciclo de vida, así como la propuesta de solución y los resultados respecto a esta.

## RECOMENDACIONES

Una vez concluida la investigación, se recomienda que:

- ✓ La Facultad procese más información sobre otros procesos inherentes al perfil y los cursos optativos, según los nuevos requerimientos que puedan surgir durante la prueba piloto del sistema.
- ✓ Se implementen otras funcionalidades que aporten comodidades, facilidades y resuelvan nuevas peticiones del cliente según necesidades.
- ✓ La Facultad solicite antes de comenzar el curso 2007 – 2008 los webservices necesarios para actualizar con datos reales la base de datos creada que soportan los subsistemas.
- ✓ La facultad conserve el documento como consulta y guía para aquellos que darán mantenimiento al sistema
- ✓ El equipo de desarrollo que dará mantenimiento al sistema SGIF implemente el módulo de administración para la seguridad de este.

**REFERENCIAS BIBLIOGRÁFICAS**

- Curso de Informatica Básica. 1997. [2007]. Disponible en:  
<http://www.monografias.com/trabajos11/curinfa/curinfa.shtml>
- (CNC)), U. D. A. C. N. D. C. Sistema de Gestión de Servicios (SGS), 2005. [febrero 20-2007].  
Disponible en: <http://www.cnc.una.py/desoft/sgs.htm>
- CÓRDOBA, U. D. Sistema Integral de Gestión Académica 2007. [enero 7-2007]. Disponible en:  
<http://www.gestion.uco.es/gestion/aplicaciones/siga>
- CORPORATION, N. PhpED Manual, 2003. [enero 10-2007]. Disponible en:  
Curso de Informatica Básica. 1997. [2007]. Disponible en:  
<http://www.monografias.com/trabajos11/curinfa/curinfa.shtml>
- EQUINOCCIAL, U. T. REGLAMENTO DEL ALUMNO, 2007. [enero 7-2007]. Disponible en:  
<http://www.ute.edu.ec/investigacion/idic/reglamento.html>
- ESEMTIA, G. El sistema MentorVox, 2004. [diciembre 20-2006]. Disponible en:  
<http://www.mentorvox.com/mentorvox.asp>
- FACTORÍA DE INTERNET S.L., S. T. D. W. S. W. C. MANUAL DE JAVASCRIPT, 2007. [mayo 2-2007]. Disponible en: <http://www.webtaller.com/manual-javascript/introduccion-javascript.php>
- LARMAN, C. UML y Patrones. 1999. 357-359 p.
- LIM, J. Librería ADOdb para PHP, 2000. [mayo 2-2007]. Disponible en:  
<http://www.lacorona.com.mx/fortiz/adodb/docs-adodb-es.htm#intro>
- MACROMEDIA, D. M. Fireworks 8 ReadMe.htm, 2002.
- MÉNDEZ, L. and A. TORRES. Sistema de Promoción y Gestión Comercial para la oficina de Transferencia Tecnológica de la Universidad de Cienfuegos, Instituto Superior Politécnico “José Antonio Echeverría”, junio 2005. p.
- MOLPECERE, A. Procesos de desarrollo: RUP, XP y FDD, 2002. [mayo 10-2007]. Disponible en:  
<http://www.javahispano.org/articles.article.action?id=76>
- NORTE, U. D. SISTEMAS DE INFORMACIÓN 2004. [enero 7-2007]. Disponible en:  
[http://www.uninorte.edu.co/acreditacion/sistemas\\_informacion.html](http://www.uninorte.edu.co/acreditacion/sistemas_informacion.html)
- OHRT, M.; A. ZMIEVSKI, et al. Smarty Manual, 2004. [mayo 2-2007]. Disponible en:  
<http://smarty.php.net/distributions/manual/es/Smarty-2.6.5-docs.pdf>

- PÉREZ, Y. and Y. SÁNCHEZ. Registro de Partos y Nacimientos para el Sistema Integral de Salud. Ciudad de la Habana, Instituto Superior Politécnico “José Antonio Echeverría”, junio 2005. p.
- PERU, F. J. U. G. Quisiera saber sobre las n-capas, 2006. [mayo 10-2007]. Disponible en: <http://www.jugperu.com/portal/modules.php?name=Forums&file=viewtopic&t=195>
- PLATA, F. D. I. U. N. D. L. Plan de estudios 2006-2007., 2006. [enero 7-2007]. Disponible en: <http://www.info.unlp.edu.ar>
- UMCC. Sistemas de Gestión Académica, 2005. [enero 7-2007]. Disponible en: <http://www.umtz.edu.cu/>
- UPM, F. G. Guía del curso Académico 2006-2007, 2006. [enero 7-2007]. Disponible en: [http://www.fi.upm.es/docs/estudios/4\\_guia\\_curso\\_06-07.pdf](http://www.fi.upm.es/docs/estudios/4_guia_curso_06-07.pdf)
- VERACUZ-MEXICO, U. D. Modernización, simplificación y mejoramiento de la eficiencia del sistema de planeación, gestión, administración y evaluación institucional., 2004. [enero 7-2007]. Disponible en: <http://www.uv.mx/universidad/doctosofi/4informe2004/10/contenido.htm>
- WIKIPEDIA, E. E. E. D. JavaScript, 2007a. [mayo 2-2007]. Disponible en: <http://es.wikipedia.org/wiki/JavaScript>
- . Smarty, 2007b. [mayo 2-2007]. Disponible en: <http://es.wikipedia.org/wiki/Smarty>

## BIBLIOGRAFÍA

1. BOGGS, W. and M. BOGGS UML with Rational Rose 2002, 2002.
2. DESARROLLOWEB. Programación en PHP, 2007. [2007]. Disponible en:  
<http://www.desarrolloweb.com/manuales/12/>
3. ESPAÑOL, M. Microsoft SQL Server 2005, 2007. [2007]. Disponible en:  
<http://www.microsoft.com/spain/sql/default.mspix>
4. GOOD, R. Ajax: El Nuevo Abordaje De Interfaz De Desarrollo De Diseño Web Del Que Todos Hablan, 2005. [2007]. Disponible en:  
[http://www.masternewmedia.org/es/2005/07/28/ajax\\_el\\_nuevo\\_abordaje\\_de.htm](http://www.masternewmedia.org/es/2005/07/28/ajax_el_nuevo_abordaje_de.htm)
5. GROUP, T. P. Capítulo 52. Problemas con bases de datos, 2001. [2007]. Disponible en:  
<http://www.php.net/manual/es/faq.databases.php>
6. HERAS, D. D. L. C. and C. Z. RODRÍGUEZ. Flash PHP y MySql Contenidos Dinámicos. 2003. 106 p.
7. HOJTSY, G.; Á. PARDO, et al. Manual de PHP, 2006. [2007]. Disponible en:  
<http://www.php.net/manual/es/>
8. JACOBSON, I.; G. BOOCH, 1. El Proceso Unificado de Desarrollo de Software., 2000. p.
9. JACOBSON, I.; G. BOOCH, 2. El Proceso Unificado de Desarrollo de Software., 2002. p.
10. JACOBSON, I.; G. BOOCH, 3. El Proceso Unificado de Desarrollo de Software., 2000. p.
11. MATANZAS, U. D. Historia 2006. [2007]. Disponible en:  
<http://www.umcc.cu/Institucion/InHistoria.html>
12. MESEGUER, P. La nueva gestión académica, 2007. [2007]. Disponible en:  
[http://www.microsoft.com/spain/enterprise/perspectivas/numero\\_7/educacion.mspix](http://www.microsoft.com/spain/enterprise/perspectivas/numero_7/educacion.mspix)
13. NAVARRA, U. D. Tutorial de JavaScript 2007. [2007]. Disponible en:  
<http://www.unav.es/cti/manuales/TutorialJavaScript/indices/>
14. OPTATIVOS, F. R. D. L. C. Asignaturas optativas básicas u obligatorias., 2006. [2006].  
Disponible en:  
[http://10.33.20.200/Documentacion/Docencia/Cursos\\_Optativos/Cursos%20do%20perfil%20Fac%20X.txt](http://10.33.20.200/Documentacion/Docencia/Cursos_Optativos/Cursos%20do%20perfil%20Fac%20X.txt)

15. RODRÍGUEZ, C. L. Ejemplo de desarrollo software utilizando la metodología RUP., 2003. [2007].  
Disponible en:  
<http://www.dsic.upv.es/asignaturas/facultad/lsi/ejemplorup/Implementacion2.html#almacen>
16. SCHMULLER, J. UML en 24 horas. 2000. p.
17. TORRES, P. L. Ingeniería del Software Orientada a Objetos y UML, 2007. [2007]. Disponible en:  
<https://pid.dsic.upv.es/C1/Material/Documentos%20Disponibles/ITI%20-%20IS%20OO%20y%20UML%20versi%C3%B3n%20Febrero%202007.ppt>
18. WIKIPEDIA., E. E. E. D. Microsoft SQL Server, 2007. [2007]. Disponible en:  
[http://es.wikipedia.org/wiki/Microsoft\\_SQL\\_Server](http://es.wikipedia.org/wiki/Microsoft_SQL_Server)
19. WIKIPEDIA. Oracle, 2007. [2007]. Disponible en: <http://es.wikipedia.org/wiki/Oracle>
20. WIKIPEDIA. PHP, 2007. [2007]. Disponible en: <http://es.wikipedia.org/wiki/PHP>
21. WIKIPEDIA. Proceso Unificado de Rational, 2007. [2007]. Disponible en:  
<http://es.wikipedia.org/wiki/RUP>

## GLOSARIO DE TÉRMINOS

**APACHE:** es un servidor HTTP de código abierto para plataformas Unix (BSD, GNU/Linux, etcétera), Windows y otras, que implementa el protocolo HTTP/1.1.

**CUN:** Caso de uso del negocio.

**CUS:** Caso de uso del sistema.

**DHTML:** Dynamic HTML.

**HTML:** HyperText Markup Language. Lenguaje usado para escribir documentos para servidores World Wide Web. Es una aplicación de la ISO Standard 8879:1986. Es un lenguaje de marcas. Los lenguajes de marcas no son equivalentes a los lenguajes de programación aunque se definan igualmente como "lenguajes". Son sistemas complejos de descripción de información, normalmente documentos, que se pueden controlar desde cualquier editor ASCII.

**HTTP:** HyperText Transfer Protocol. Protocolo de Transferencia de Hipertextos. Modo de comunicación para solicitar páginas Web.

**Herramientas CASE:** Herramientas utilizadas para el desarrollo de proyectos de Ingeniería de Software.

**Hardware:** Componentes electrónicos, tarjetas, periféricos y equipo que conforman un sistema de computación; se distinguen de los programas (software) porque son tangibles.

**Internet:** Sistema de redes de computación ligadas entre sí, con alcance mundial, que facilita servicios de comunicación de datos como registro remoto, transferencia de archivos, correo electrónico y grupos de noticias. Internet es una forma de conectar las redes de computación existentes que amplía en gran medida el alcance de cada sistema participante.

**Macromedia Dreamweaver MX:** Herramienta para el desarrollo de aplicaciones Web de Macromedia. Combina en un único entorno de desarrollo accesible y potente las reconocidas herramientas de presentación visual de Dreamweaver, las características de rápido desarrollo de aplicaciones Web de Dreamweaver UltraDev y ColdFusion Studio, y el extenso soporte de edición de código de HomeSite. Ofrece una completa solución abierta para las tecnologías Web y estándares de hoy, incluyendo la accesibilidad y servicios Web.

**Microsoft:** Compañía que manufactura los sistemas de operación DOS y Windows.

**MySQL:** Es un sistema de gestión de bases de datos relacional que cuentan con todas las características de un motor de BD comercial: transacciones atómicas, triggers, replicación, llaves foráneas entre otras. Su ingeniosa arquitectura lo hace extremadamente rápido y fácil de personalizar.

**PHP:** *PHP: Hypertext Preprocessor.* Es un ambiente script del lado del servidor que permite crear y ejecutar aplicaciones Web dinámicas e interactivas. Con PHP se pueden combinar páginas HTML y scripts. Con el objetivo de crear aplicaciones potentes.

**PostgreSQL:** es un Sistema de Gestión de Bases de Datos Objeto-Relacionales (ORDBMS) libre.

**RUP:** *Rational Unified Process* (Proceso Unificado de desarrollo). Metodología para el desarrollo de Software.

**Software:** Programas de sistema, utilerías o aplicaciones expresados en un lenguaje de máquina.

**SGBD:** *Sistema de Gestión de Bases de Datos.* Es el software que permite la utilización y/o la actualización de los datos almacenados en una (o varias) base(s) de datos por uno o varios usuarios desde diferentes puntos de vista y a la vez.

**UML:** *Unified Modeling Language.* Es una notación estándar para modelar objetos del mundo real como primer paso en el desarrollo de programas orientados a objetos. Es un lenguaje para visualizar, especificar, construir y documentar los artefactos de un sistema de software.

**WEB (WWW):** Red de documentos HTML intercomunicados y distribuidos entre servidores del mundo entero.