

Universidad de las Ciencias Informáticas

Facultad 1



*Título: “Sistema de Gestión de los Cursos Optativos
en la Facultad 1.”*

*Trabajo de Diploma para optar por el título de Ingeniero
en Ciencias Informáticas.*

Autor: Alejandro Llorente Chaviano.

Tutor: Ing. Roberlán Rodríguez Sánchez.

Ciudad de La Habana, Cuba

Junio 2009

“Año del 50 Aniversario del Triunfo de la Revolución”.

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

Sir Francis.

DECLARACIÓN DE AUTORÍA

Declaro que soy el único autor de este trabajo y autorizo a la Universidad de las Ciencias Informáticas (UCI) para que haga el uso que estimen pertinente con el mismo.

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

Firma del Autor

Alejandro Llorente Chaviano.

Firma del Tutor

Ing. Roberlán Rodríguez Sánchez.

Quiero agradecer:

A nuestra revolución, a Raúl y a Fidel.

Al tutor Ing. Roberlán Rodríguez Sánchez que con tanto esfuerzo y dedicación supo guiarme en el desarrollo de este trabajo.

Otros Agradecimientos:

A Yeni por el significado de su compañía, apoyo y amor en estos 3 años, gracias mi tática.

A mis abuelos Aidé Días y Pedro Chaviano, ellos siempre me enseñaron algo nuevo de su vasta experiencia y a ir siempre por el camino correcto.

A mi tía Malvelis en especial porque siempre ha estado pendiente de mí como una madre.

A mi hermano, que siempre ha seguido mis pasos, espero obtenga en el futuro logros más grandes que los míos.

De forma general toda la mi familia que no ha tenido reparos en ayudarme a lograr alcanzar mis metas.

A mis tíos Raquel y Risel, y a mi prima Yari que me apoyaron siempre y me dieron ánimos a seguir adelante.

A mis suegros Delia y Juan por ser tan bellas personas.

A los amigos de la secundaria, el pre y a los de la universidad, en especial Yanisley, Alieski, Yunieski, Leordan y Danny. Y muy especial a Reynaldo por su incondicional apoyo.

A mis abuelos, Aidé Días y Pedro Chaviano por guiarme por el buen camino, gracias a ellos he obtenido lo logrado....

De su nieto Alejandro Llorente Chaviano.

A mi madre, Mayci Chaviano y a mi padrastro Guillermo Rojas por el empeño, paciencia y dedicación con que me han conducido en la vida....

De su hijo Alejandro Llorente Chaviano

Resumen

En la Universidad de las Ciencias Informáticas fluye información relacionada con los cursos optativos, ofertados a los estudiantes durante la carrera, la cual es procesada mediante múltiples vías y por el personal indicado. En la Facultad 1, estos datos se controlan y procesan de manera manual, con el objetivo de obtener estadísticas según necesidades o peticiones de los implicados y directivos, provocando en ocasiones: gasto de tiempo, mano de obra y algunos incumplimientos en la entrega de información. La matrícula en los cursos optativos se hace en la Secretaría, que dada la convocatoria hecha por correo a los estudiantes con fecha límite, los mismos pasan por dicho local 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, además en ciertas ocasiones los estudiantes tienen que esperar bastante tiempo para matricularse puesto que la secretaria tiene que recoger sus datos de forma manual y sumado que los estudiantes suelen acudir en grupos o que en ocasiones tienen que ir varias veces, el proceso se torna lento, provocando que la matrícula no satisfaga lo planificado o previsto inicialmente. 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. El contacto con los profesores para la coordinación del curso también es un tanto complicado. Se hace vía correo electrónico lo que provoca que el profesor dependa de este medio y luego así mismo él se comunique con sus alumnos. 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 los Cursos Optativos de la Facultad 1”, 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. 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 y cumpliendo los objetivos propuestos.

INTRODUCCIÓN	1
FUNDAMENTACIÓN TEÓRICA	6
1.1 INTRODUCCIÓN.....	6
1.2 ESTADO DEL ARTE.....	6
1.2.1 <i>Conceptos fundamentales</i>	6
1.2.2 <i>Actualidad del trabajo.</i>	7
1.2.3 <i>Beneficios de los sistemas de gestión de información.</i>	11
1.3 TECNOLOGÍAS, HERRAMIENTAS Y METODOLOGÍA A UTILIZAR.	12
1.3.1 <i>Lenguajes de programación para la Web.</i>	12
1.3.2 <i>Frameworks en PHP.</i>	15
1.3.3 <i>Sistemas Gestores de Base de Datos.</i>	17
1.3.4 <i>Metodologías de desarrollo de software: Rational Unified Process (RUP).</i>	19
1.3.5 <i>Lenguaje de Modelado: UML</i>	19
1.3.6 <i>Herramientas Case: Visual Paradigm</i>	20
1.4 CONCLUSIONES.....	21
PROPUESTA DEL SISTEMA.....	22
2.1. INTRODUCCIÓN.	22
2.2. REGLAS DEL NEGOCIO.	22
2.2.1. <i>Descripción de los procesos del negocio.</i>	23
2.3. REQUERIMIENTOS FUNCIONALES.....	31
2.4. REQUISITOS NO FUNCIONALES.	34
2.5. DEFINICIÓN DE LOS PROCESOS DEL SISTEMA.....	35
2.5.1. <i>Descripción del sistema propuesto.</i>	35
2.6. CONCLUSIONES.	45
ANÁLISIS Y DISEÑO	46
3.1 INTRODUCCIÓN.....	46
3.2 ANÁLISIS.....	46
3.3 DISEÑO.....	48
3.4 DESCRIPCIÓN DE LAS TABLAS DE LA BASE DE DATO	51
3.5 ESTIMACIÓN DE ESFUERZO.	55
3.5.1 <i>Cálculo de Puntos de Casos de Uso sin ajustar.</i>	55
3.5.2 <i>Cálculo de los Puntos de Casos de Uso ajustados.</i>	57
3.5.3 <i>Estimación de esfuerzo a través de los puntos de casos de uso.</i>	59
3.5.4 <i>Calcular el esfuerzo de todo el proyecto.</i>	59
3.6 CONCLUSIONES.....	60
IMPLEMENTACIÓN Y PRUEBA.....	61
4.1. INTRODUCCIÓN	61
4.2. IMPLEMENTACIÓN	61
4.2.1. <i>Diagrama de Despliegue.</i>	61
4.2.2. <i>Diagramas de Componentes</i>	62

4.3.	MODELO DE PRUEBA	64
4.3.1.	<i>Casos de Prueba</i> -----	65
4.4.	CONCLUSIONES	68
	CONCLUSIONES GENERALES	69
	RECOMENDACIONES.....	70
	REFERENCIAS BIBLIOGRÁFICAS	71
	BIBLIOGRAFÍA.....	72
	GLOSARIO DE TÉRMINOS.....	73
	ANEXOS	74
	ANEXO1. DESCRIPCIÓN DE LOS CASOS DE USO DEL SISTEMA.....	74
	ANEXO2. DIAGRAMAS DE CLASES DEL ANÁLISIS.....	88
	ANEXO3. DIAGRAMAS DE CLASES DEL DISEÑO.	91
	ANEXO4. DIAGRAMAS DE COMPONENTES.	96

TABLA 1: DESCRIPCIÓN DE LOS ACTORES DEL NEGOCIO.....	23
TABLA 2: DESCRIPCIÓN DE LOS TRABAJADORES DEL NEGOCIO.....	23
TABLA 3: DESCRIPCIÓN TEXTUAL DEL CUN CREAR MATRÍCULA.....	24
TABLA 4: DESCRIPCIÓN TEXTUAL DEL CUN CANCELAR MATRÍCULA.....	26
TABLA 5: DESCRIPCIÓN TEXTUAL DEL CUN SOLICITAR INFORMACIÓN.....	27
TABLA 6: DESCRIPCIÓN DE LOS ACTORES DEL SISTEMA.....	36
TABLA 7: PAQUETES DE CUS.....	36
TABLA 8: DESCRIPCIÓN TEXTUAL DEL CUS AUTENTICAR USUARIO.....	37
TABLA 9: DESCRIPCIÓN TEXTUAL DEL CUS GESTIONAR MATRÍCULA.....	39
TABLA 15: DESCRIPCIÓN TEXTUAL DEL CUS GESTIONAR CO.....	41
TABLA 16: DESCRIPCIÓN DE LA TABLA AÑOS/CO.....	51
TABLA 17: DESCRIPCIÓN DE LA TABLA AÑOS.....	52
TABLA 18: DESCRIPCIÓN DE LA TABLA MODALIDAD.....	52
TABLA 19: DESCRIPCIÓN DE LA TABLA PERFIL.....	52
TABLA 20: DESCRIPCIÓN DE LA TABLA DISCIPLINA.....	52
TABLA 21: DESCRIPCIÓN DE LA TABLA CO.....	53
TABLA 22: DESCRIPCIÓN DE LA TABLA MATRÍCULA.....	53
TABLA 23: DESCRIPCIÓN DE LA TABLA ROL.....	53
TABLA 24: DESCRIPCIÓN DE LA TABLA ROL/USUARIO.....	54
TABLA 25: DESCRIPCIÓN DE LA TABLA USUARIO.....	54
TABLA 26: DESCRIPCIÓN DE LA TABLA PROFESOR.....	54
TABLA 27: DESCRIPCIÓN DE LA TABLA ESTUDIANTE.....	54
TABLA 28: FACTOR DE PESOS DE LOS ACTORES SIN AJUSTAR.....	55
TABLA 29: FACTOR DE PESOS DE LOS CASOS DE USO SIN AJUSTAR.....	56
TABLA 30: FACTOR DE COMPLEJIDAD TÉCNICA.....	57
TABLA 31: FACTOR DE AMBIENTE.....	58
TABLA 32: ESFUERZO DEL PROYECTO.....	59
TABLA 33: CASO DE PRUEBA: CREAR CURSO OPTATIVO.....	65
TABLA 34: CASO DE PRUEBA: MATRICULARSE EN CURSO OPTATIVO.....	66
TABLA 35: CASO DE PRUEBA: MATRÍCULA FORZADA.....	67
TABLA 10: DESCRIPCIÓN TEXTUAL DEL CUS GESTIONAR PERFIL.....	74

TABLA 11: DESCRIPCIÓN TEXTUAL DEL CUS GESTIONAR DISCIPLINA.....	77
TABLA 12: DESCRIPCIÓN TEXTUAL DEL CUS GESTIONAR AÑO.....	79
TABLA 13: DESCRIPCIÓN TEXTUAL DEL CUS GESTIONAR ADMINISTRADOR.....	82
TABLA 14: DESCRIPCIÓN TEXTUAL DEL CUS MATRÍCULA FORZADA.	85

FIGURA 1: DIAGRAMA DE CASOS DE USO DEL NEGOCIO.....	24
FIGURA 2: DIAGRAMA DE ACTIVIDADES CREAR MATRÍCULA.....	28
FIGURA 3: DIAGRAMA DE ACTIVIDADES CANCELAR MATRÍCULA.....	29
FIGURA 4: DIAGRAMA DE ACTIVIDADES SOLICITAR INFORMACIÓN.....	30
FIGURA 5: DIAGRAMA DEL MODELO DE OBJETOS.....	31
FIGURA 6: DIAGRAMA DE CASOS DE USO DEL SISTEMA.....	37
FIGURA 7: DIAGRAMA DE CLASES DEL ANÁLISIS AUTENTICAR USUARIO.....	46
FIGURA 12: DIAGRAMA DE CLASES DEL ANÁLISIS GESTIONAR CO.....	47
FIGURA 13: DIAGRAMA DE CLASES DEL ANÁLISIS GESTIONAR MATRÍCULA.....	47
FIGURA 15: DIAGRAMA DE CLASES DEL DISEÑO AUTENTICAR USUARIO.....	48
FIGURA 20: DIAGRAMA DE CLASES DEL DISEÑO GESTIONAR CO.....	49
FIGURA 21: DIAGRAMA DE CLASES DEL DISEÑO GESTIONAR MATRÍCULA.....	50
FIGURA 23: DIAGRAMA ENTIDAD-RELACIÓN.....	51
FIGURA 24: DIAGRAMA DE DESPLIEGUE.....	61
FIGURA 25: DIAGRAMA DE COMPONENTES AUTENTICAR USUARIO.....	62
FIGURA 28: DIAGRAMA DE COMPONENTES GESTIONAR CO.....	63
FIGURA 38: DIAGRAMA DE COMPONENTES GESTIONAR MATRÍCULA.....	63
FIGURA 40: REPRESENTACIÓN DE LA PRUEBA DE CAJA NEGRA.....	64
FIGURA 8: DIAGRAMA DE CLASES DEL ANÁLISIS GESTIONAR AÑOS.....	88
FIGURA 9: DIAGRAMA DE CLASES DEL ANÁLISIS GESTIONAR DISCIPLINA.....	88
FIGURA 10: DIAGRAMA DE CLASES DEL ANÁLISIS GESTIONAR PERFIL.....	89
FIGURA 11: DIAGRAMA DE CLASES DEL ANÁLISIS GESTIONAR ADMINISTRADOR.....	89
FIGURA 14: DIAGRAMA DE CLASES DEL ANÁLISIS GESTIONAR MATRÍCULA FORZADA.....	90
FIGURA 16: DIAGRAMA DE CLASES DEL DISEÑO GESTIONAR AÑOS.....	91
FIGURA 17: DIAGRAMA DE CLASES DEL DISEÑO GESTIONAR DISCIPLINA.....	92
FIGURA 18: DIAGRAMA DE CLASES DEL DISEÑO GESTIONAR PERFIL.....	93
FIGURA 19: DIAGRAMA DE CLASES DEL DISEÑO GESTIONAR ADMINISTRADOR.....	94
FIGURA 22: DIAGRAMA DE CLASES DEL DISEÑO GESTIONAR MATRÍCULA FORZADA.....	95
FIGURA 26: DIAGRAMA DE COMPONENTES GESTIONAR AÑOS.....	96
FIGURA 27: DIAGRAMA DE COMPONENTES GESTIONAR AÑOS-CO.....	96
FIGURA 29: DIAGRAMA DE COMPONENTES GESTIONAR DISCIPLINA.....	97

FIGURA 30: DIAGRAMA DE COMPONENTES GESTIONAR ESTUDIANTE-CO.....	97
FIGURA 31: DIAGRAMA DE COMPONENTES GESTIONAR ESTUDIANTE.....	98
FIGURA 32: DIAGRAMA DE COMPONENTES GESTIONAR PERFIL.	98
FIGURA 33: DIAGRAMA DE COMPONENTES GESTIONAR PROFESOR.....	99
FIGURA 34: DIAGRAMA DE COMPONENTES GESTIONAR ROL.	99
FIGURA 35: DIAGRAMA DE COMPONENTES GESTIONAR USUARIO.....	100
FIGURA 36: DIAGRAMA DE COMPONENTES GESTIONAR USUARIO-ROL.	100
FIGURA 37: DIAGRAMA DE COMPONENTES HABILITAR.	101
FIGURA 39: DIAGRAMA DE COMPONENTES MATRÍCULA FORZADA.....	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, avanza extraordinariamente.

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

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.

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 las 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.

En la mayoría de 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. 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. La forma de evaluación es mediante prueba de suficiencia o recibiendo el curso

durante un período determinado y realizando un trabajo final como muestra de los conocimientos adquiridos.

La Universidad de las Ciencias Informáticas, es un centro que se caracteriza por ofertar una gran cantidad de cursos optativos sobre distintas materias entre las que encontramos: 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.

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, los estudiantes pasan por dicho local 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, además en ciertas ocasiones los estudiantes tienen que esperar bastante tiempo para matricularse puesto que la secretaria tiene que recoger sus datos de forma manual y sumado que los estudiantes suelen acudir en grupos o que en ocasiones tienen que ir varias veces, el proceso se torna lento provocando que la matrícula no satisfaga lo planificado o previsto inicialmente. 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. El contacto con los profesores para la coordinación del curso también es un tanto complicado. Se hace vía correo electrónico lo que provoca que el profesor dependa de este medio y luego así mismo él se comunique con sus alumnos.

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

Al no contar con un sistema informatizado que permita la gestión de los cursos optativos que se imparten en la Facultad 1, se plantea la siguiente **situación problémica**: la gestión de los cursos optativos en la Facultad 1, acarrear procesos que involucran tanto a profesores como directivos de la Facultad, todo esto requiere de intercambio físico entre los involucrados, lo que ocasiona demoras y gastos de recursos.

Luego de analizar la situación problémica, surge el siguiente **problema científico**: ¿Cómo facilitar de forma eficiente, segura y rápida el proceso de gestión de la información relacionada con los cursos optativos en la Facultad 1 de la Universidad de las Ciencias Informáticas? Este problema se enmarca en el

siguiente **objeto de estudio**: proceso de gestión de información académica en la Facultad 1 de la Universidad de las Ciencias Informáticas. Para resolver el problema se propone el siguiente **objetivo general**: desarrollar un sistema informático que gestione los procesos relacionados con los cursos optativos y el proceso de matrícula de los estudiantes en los mismos, en la Facultad 1 de la Universidad de las Ciencias Informáticas.

El objetivo delimita el siguiente **campo de acción**: proceso de gestión de la información relacionada con los cursos optativos que se imparten en la Facultad 1 de la Universidad de las Ciencias Informáticas y la información inherente a la matrícula de los estudiantes en los mismos.

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

- Realizar el marco teórico de la investigación.
- Modelar los procesos de gestión de los cursos optativos y matrícula de los estudiantes en la Facultad 1 de la Universidad de las Ciencias Informáticas.
- Implementar el sistema de gestión de cursos optativos.
- Probar el sistema desarrollado.

Para dar cumplimiento a los objetivos trazados se desarrollarán las siguientes **tareas**:

- Realizar entrevistas para conocer cómo se desarrolla el proceso de gestión de los cursos optativos y el proceso de matrícula de los estudiantes en la Facultad 1 de la Universidad de las Ciencias Informáticas.
- Identificar los problemas existentes en dichos procesos.
- Realizar un estudio de los sistemas similares existentes tanto a nivel nacional como internacional.
- Identificar las herramientas, tecnologías y metodologías más adecuadas para la implementación del sistema.
- Definir los requisitos funcionales y no funcionales del sistema propuesto.
- Realizar las pruebas de caja negra de los artefactos generados.

Idea a defender

Con la implementación del sistema para la gestión de los cursos optativos en la Facultad 1, se lograrán mejorar los procesos relacionados con la creación y matrícula de los cursos optativos, manteniendo segura la actualización de la información inherente a dichos procesos.

Para dar cumplimiento a las actividades programadas en el trabajo se combinan diferentes métodos y técnicas en la búsqueda y procesamiento de la información.

Los **Métodos Teóricos** utilizados fueron:

- Análítico - Sintético: Se utiliza en la definición de las funcionalidades.
- Análisis sistémico: Durante el desarrollo del análisis y diseño mediante la elaboración de los diagramas de clases.
- Análisis histórico – lógico: es el estudio sobre los antecedentes del proceso de gestión de la información así como el análisis de cómo funciona este proceso.

Los **Métodos Empíricos** utilizados en la investigación fueron:

- Método de Entrevista: Se utiliza para obtener información relacionada con la gestión de los CO.
- Método de Observación: Se utiliza para visualizar de forma personal la información obtenida mediante las entrevistas.

Aportes prácticos esperados del trabajo

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

- Disminuir 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 1, permitiendo que sea más rápido, confiable y seguro.
- Actualizar, controlar y procesar información sobre los cursos de forma rápida.
- Con lo anteriormente dicho se pretende que aumente la calidad del trabajo en la Facultad 1.

A modo de resumen se espera obtenga un software que permita:

- Ordenar los datos e interactuar dinámicamente con la información existente.
- Tener disponible la información relacionada con los cursos optativos de la Facultad 1 en una Web.
- Hacer accesible y actualizable la información de los cursos optativos en la Facultad 1, desde cualquier punto de la Universidad.

El contenido del presente trabajo está dividido en 4 capítulos distribuidos de la manera siguiente: En el capítulo 1 se trata la fundamentación teórica, incluye un profundo estado del arte abordando temas como

las últimas tendencias, técnicas, metodologías y programas que se usan en la actualidad para desarrollar aplicaciones web. Todo esto a nivel internacional y nacional, enfocándonos en nuestra Universidad como principal desarrollador de software de nuestro país en estos momentos.

El capítulo 2 describe el sistema que proponemos para darle solución a la problemática actual que da origen al mismo. En primer lugar describe ampliamente esta situación, especificando como se desarrolla el proceso de gestión de cursos optativos en la Facultad 1 de la Universidad de las Ciencias Informáticas y los problemas que trae consigo el empleo de estos mecanismos. Se especifica el modelo de negocio y se recogen las características básicas del sistema expresadas en funcionalidades y los requerimientos no funcionales como apariencia, rendimiento, confiabilidad, entre otras. Por último se definen los casos de uso del sistema, así como los actores y la explicación detallada de cada caso de uso.

El capítulo 3 recoge el análisis y diseño de la solución propuesta. Esto incluye la definición del modelo del análisis o las clases del análisis. Se describen las clases del diseño, especificando sus atributos y operaciones.

El capítulo 4 está dedicado a la fase implementación y pruebas. Aquí se ilustra el diagrama de despliegue y de componentes como parte de la implementación. En el caso de la fase de pruebas se describen los casos de pruebas que se emplearon en la misma.

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.

La Informática en la gestión de la información.

La informática es la disciplina que estudia el tratamiento automático de la información utilizando dispositivos electrónicos y sistemas computacionales, definida además como el procesamiento de información en forma automática. Para ello los sistemas informáticos deben captar la información, que esta sea procesada o tratada para luego mostrar o transmitir los resultados. La informática se aplica a

Capítulo 1: *Fundamentación Teórica.*

diversas áreas, gestión de negocio, almacenamiento de información, monitorización y control de procesos de investigación, desarrollo de juegos, diseño computarizado, aplicaciones/herramientas multimedia.

La informática pretende brindar sistemas de gestión cada vez más eficientes a través de los cuales se logre la integración coherente de grandes cantidades de datos, al permitir la organización y calidad de los mismos, con lo cual se afirma que durante los últimos años con el desarrollo de herramientas y aplicaciones informáticas se han realizado grandes avances en la manipulación de grandes cantidades de datos.

SIG (Sistemas de Gestión de la Información.)

La definición de Sistema para la Gestión de Información, desde el punto de vista de la lógica de Marketing, resulta particularmente interesante. En efecto, sugiere que sea el propio sistema (y no el humano) el sujeto de la Gestión. Una definición más apropiada podría ser la de "sistema de soporte a la Gestión de Información" ya que, en realidad, son las estrategias de Comunicación las que realmente llevan a gestionar información de forma efectiva; los sistemas informáticos pueden a lo sumo proporcionar las herramientas necesarias, o bien incluir servicios de soporte a la toma de decisiones por lo que a la Gestión de Información se refiere. Muchos sistemas proporcionan diferentes niveles de acceso dependiendo del usuario. El acceso a los Sistema para la Gestión de Información se realiza generalmente a través del navegador web, y a veces se requiere el uso de FTP para subir contenidos. Los sistemas de gestión de información están concebidos para capturar, actualizar, integrar, consultar y analizar información pertinente a la organización a la cual pertenecen.

1.2.2 Actualidad del trabajo.

Actualmente, en todo 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 para permitir su capacitación. Los softwares de gestión de información académica son muy utilizados en cualquier centro de enseñanza gracias a las comodidades que ellos proporcionan al claustro de profesores y a los estudiantes.

Capítulo 1: Fundamentación Teórica.

A Nivel Mundial

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.

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.

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.

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

Capítulo 1: Fundamentación Teórica.

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.

A Nivel Nacional.

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). Pero existen diversos sistemas que igualmente gestionan información entre los que destacan:

Servicio de Consultoría de Sistema de Gestión de Información **(SCSGI)**.

El servicio está basado en la necesidad de un reposicionamiento de la información como base del conocimiento, que constituye el eje central del cambio hacia una nueva sociedad con empresas perfeccionadas. Este aborda las acciones y beneficios del diseño e implementación del Sistema de Gestión de Información, que permiten diagnosticar y organizar los recursos de información, determinar el valor y disminuir los costo de la información; además de evaluar la efectividad actual de la gestión de información, se elaboran los mapas de flujos de información por subsistemas y el general; a la vez que permite diseñar, organizar y formalizar la Gestión Documental y el trabajo de la Unidad de Información, resultados que se perciben a través de la Intranet Corporativa de la organización. Propone un plan de medidas que permite a la empresa ascender a un estadio superior.

Un aspecto novedoso de este trabajo es que inserta la Gestión Documental, en apoyo al Decreto Ley 221 del 8 de agosto del 2001 del Sistema Nacional de Archivos y contribuye al cumplimiento de la 297 del Control Interno y a la 187 del Perfeccionamiento Empresarial.

El proyecto se desarrolla a través de cuatro etapas: Diagnóstico preliminar, Sistema de Gestión Documental, Gestión de la Unidad de Información, Arquitectura de contenidos (Intranet). El sistema facilita a los clientes internos de la organización la información adecuada, en la forma correcta, para la

Capítulo 1: Fundamentación Teórica.

persona indicada, al costo adecuado, en el tiempo oportuno, en el lugar apropiado, para tomar la acción correcta.

Redes y Conocimiento en Acción.

Este sistema mantiene informados a los especialistas e investigadores sobre los avances científicos y tecnológicos en los diversos campos. Las tecnologías de la información extienden los documentos de ciencia y tecnología en formato electrónico y las bases de datos de registro documental. Para aprovechar este inmenso caudal de conocimientos se requieren herramientas para analizar grandes volúmenes de información.

En la UCI

En la *Universidad de las Ciencias Informáticas (UCI)* se cuenta 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 que se realicen de forma rápida la matrícula de los estudiantes y el proceso de gestión de los CO, 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 los cursos optativos, ejemplo: 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 Jefe 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 Jefe 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.

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.

Sistema del Corte, Sistema de Asistencia y Sistema de Evaluaciones.

Estos sistemas son creados por la Dirección de Informatización de la UCI en el año 2003-2004 con el objetivo de garantizar la gestión académica de la universidad y satisfacer las necesidades de información de sus usuarios.

Los Sistemas del Corte y de Asistencia brindan información que pueden ser consultados a nivel de facultad, año, grupo, provincia, municipio, asignatura, sexo, raza, vía de ingreso y procedencia.

El Sistema del Corte muestra la cantidad de estudiantes que hay para cada uno de los tipos de evaluaciones (Bien, Regular y Mal).

Por su parte el Sistema de Evaluaciones pone a disposición del usuario los siguientes datos:

Cantidad de estudiantes sin mundial por facultad.

Cantidad de estudiantes sin mundial por provincia.

Listado de estudiantes por grupo y facultad que no tienen evaluaciones insertadas en las asignaturas obligatorias.

Listado de estudiantes con 5 en todas las asignaturas.

Listado de estudiantes con 4 y/o 5 en todas las asignaturas.

Listado de estudiantes con 1 asignatura suspensa.

Listado de estudiantes con 2 asignaturas suspensas.

Listado de estudiantes con 3 o más asignaturas suspensas.

1.2.3 Beneficios de los sistemas de gestión de información.

Entre los principales beneficios que proporcionan los sistemas de gestión de información tenemos:

- Velocidad de procesamiento de la información y obtención de resultados.
- Consultas de la información.
- Productividad del trabajo.
- Exactitud y consistencia de los datos.
- Interrelación de las áreas con el sistema.
- Seguridad de la información.

Existen ciertos sistemas de gestión de la información que constituyen la puesta en práctica de un conjunto de teorías e investigaciones sobre el tema, es decir, son el resultado del trabajo de ingenieros, los cuales ocupan un papel importante en el progreso tecnológico e investigativo.

1.3 Tecnologías, herramientas y metodología a utilizar.

Como **propuesta de solución** se tiene desarrollar una aplicación web que gestione la información relacionada con los cursos optativos. 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.

1.3.1 Lenguajes de programación para la Web.

Actualmente existen diferentes lenguajes de programación para desarrollar en la Web, estos han ido surgiendo debido a las tendencias y necesidades de las plataformas. Desde los inicios de Internet, fueron surgiendo diferentes demandas por los usuarios y se dieron soluciones mediante lenguajes estáticos. A medida que paso el tiempo, las tecnologías fueron desarrollándose y surgieron nuevos problemas a dar solución. Esto dio lugar a desarrollar lenguajes de programación para la Web dinámicos, que permitieran interactuar con los usuarios y utilizaran sistemas de Bases de Datos

Lenguaje HTML

Desde el surgimiento de Internet se han publicado sitios Web gracias al lenguaje *HTML*. Es un lenguaje estático para el desarrollo de sitios. Desarrollado por el World Wide Web Consortium ([W3C](#)).

Ventajas:

- Sencillo que permite describir hipertexto.
- Texto presentado de forma estructurada y agradable.
- No necesita de grandes conocimientos cuando se cuenta con un editor de páginas Web o WYSIWYG.
- Archivos pequeños.
- Despliegue rápido.

- Lenguaje de fácil aprendizaje.
- Lo admiten todos los exploradores.

Desventajas:

- Lenguaje estático.
- La interpretación de cada navegador puede ser diferente.
- El diseño es más lento.
- Las etiquetas son muy limitadas.

Lenguaje Java Script

Este es un lenguaje interpretado, no requiere compilación. Fue creado por Brendan Eich en la empresa Netscape Communications. Utilizado principalmente en páginas Web. Es similar a Java, aunque no es un lenguaje orientado a objetos, el mismo no dispone de herencias.

Ventajas:

- Lenguaje de scripting seguro y fiable.
- Los script tienen capacidades limitadas, por razones de seguridad.
- El código Java script se ejecuta en el cliente.

Desventajas:

- Código visible por cualquier usuario.
- El código debe descargarse completamente.

Lenguaje PHP

Es un lenguaje de programación utilizado para la creación de sitio Web. PHP es un lenguaje de script interpretado en el lado del servidor utilizado para la generación de páginas Web dinámicas, embebidas en páginas HTML y ejecutadas en el servidor. PHP no necesita ser compilado para ejecutarse. Para su funcionamiento necesita tener instalado Apache o IIS con las librerías de PHP. La mayor parte de su

Capítulo 1: Fundamentación Teórica.

sintaxis ha sido tomada de C, Java y Perl con algunas características específicas. Los archivos cuentan con la extensión (php).

Ventajas:

- Muy fácil de aprender.
- Se caracteriza por ser un lenguaje muy rápido.
- Soporta en cierta medida la orientación a objeto. Clases y herencia.
- Es un lenguaje multiplataforma: Linux, Windows, entre otros.
- Capacidad de conexión con la mayoría de los manejadores de base de datos: MySQL, PostgreSQL, Oracle, MS SQL Server, entre otras.
- Capacidad de expandir su potencial utilizando módulos.
- Posee documentación en su página oficial la cual incluye descripción y ejemplos de cada una de sus funciones.
- Es libre, por lo que se presenta como una alternativa de fácil acceso para todos.
- Incluye gran cantidad de funciones.
- No requiere definición de tipos de variables ni manejo detallado del bajo nivel.

Desventajas:

- Se necesita instalar un servidor web.
- Todo el trabajo lo realiza el servidor y no delega al cliente. Por tanto puede ser más ineficiente a medida que las solicitudes aumenten de número.
- La legibilidad del código puede verse afectada al mezclar sentencias HTML y PHP.
- La programación orientada a objetos es aun muy deficiente para aplicaciones grandes.
- Dificulta la modularización.
- Dificulta la organización por capas de la aplicación.

PHP es el seleccionado porque además de las ventajas que proporciona y pese a sus desventajas es un lenguaje que brinda una gran seguridad. Además PHP no obliga a quien lo usa a seguir una determinada metodología a la hora de programar, y aunque este dirigido a alguna en particular, el programador puede

aplicar en su trabajo cualquier técnica de programación y/o desarrollo que le permita escribir código ordenado, estructurado y manejable.

1.3.2 Frameworks en PHP.

En la actualidad existen más de 40 frameworks disponibles en la comunidad de PHP por lo que se hace cada día más difícil decidir qué Framework funciona mejor, sobre todo porque cada uno de ellos ofrece diferentes funcionalidades. Entre los más populares se encuentran:

WACT - Web Application Toolkit:

WACT tiene las cosas básicas necesarias de un frameworks, como MVC, objetos de la base de datos y un motor de plantillas. WACT tiene uno de los mejores motores de la plantilla que le permite incluso, crear sus propios códigos de plantilla para añadir cierta funcionalidad. No tiene módulos adicionales, y definitivamente carece de algunas cosas importantes, como Ajax, almacenamiento en caché, un módulo de autenticación y más.

Prado:

Prado es un frameworks muy singular, ofrece algunas características muy interesantes aunque no viene con soporte incorporado para MVC, pero tiene todo tipo de componentes para manejar la mayoría de las tareas habituales de PHP.

PHP on Trax:

La documentación es bastante escasa, por lo que es difícil de determinar lo que ofrece este frameworks, pero su documentación hacen mención de Active Record, por lo que probablemente las ofertas Active Record, al igual que RoR. Incluye soporte incorporado para Ajax. Le faltan algunas cosas, como un módulo de autenticación y un módulo de almacenamiento en caché. No tiene módulos adicionales, por lo que es un todo en uno en este momento.

Symfony

Symfony es el seleccionado para nuestro trabajo puesto que es un completo framework diseñado para optimizar el desarrollo de las aplicaciones web mediante algunas de sus principales características. Para empezar, separa la lógica de negocio, la lógica de servidor y la presentación de la aplicación web. Proporciona varias herramientas y clases encaminadas a reducir el tiempo de desarrollo de una aplicación web compleja. Además, automatiza las tareas más comunes, permitiendo al desarrollador dedicarse por

Capítulo 1: *Fundamentación Teórica.*

completo a los aspectos específicos de cada aplicación. El resultado de todas estas ventajas es que no se debe reinventar la rueda cada vez que se crea una nueva aplicación web.

Symfony está desarrollado completamente con PHP 5. Ha sido probado en numerosos proyectos reales y se utiliza en sitios web de comercio electrónico de primer nivel. Symfony es compatible con la mayoría de gestores de bases de datos, como MySQL, PostgreSQL, Oracle y Microsoft SQL Server. Se puede ejecutar tanto en plataformas *nix como en plataformas Windows.

Symfony se diseñó para que se ajustara a los siguientes requisitos:

- Fácil de instalar y configurar en la mayoría de plataformas.
- Independiente del sistema gestor de bases de datos
- Sencillo de usar en la mayoría de casos, pero lo suficientemente flexible como para adaptarse a los casos más complejos.
- Basado en la premisa de “convenir en vez de configurar”, en la que el desarrollador solo debe configurar aquello que no es convencional.
- Sigue la mayoría de mejores prácticas y patrones de diseño para la web.
- Preparado para aplicaciones empresariales y adaptables a las políticas y arquitecturas propias de cada empresa, además de ser lo suficientemente estable como para desarrollar aplicaciones a largo plazo.
- Código fácil de leer que incluye comentarios de phpDocumentor y que permite un mantenimiento muy sencillo.
- Fácil de extender, lo que permite su integración con las librerías de otros fabricantes

Symfony está basado en un patrón clásico del diseño web conocido como arquitectura MVC, que está formado por 3 niveles:

- El modelo representa la información con la que trabaja la aplicación, es decir, su lógica de negocio.
- La vista transforma el modelo en una página web que permite al usuario interactuar con ella.
- El controlador se encarga de procesar las interacciones del usuario y realiza los cambios apropiados en el modelo o en la vista.

La arquitectura MVC separa la lógica de negocio (el modelo) y la presentación (la vista) por lo que se consigue un mantenimiento más sencillo de las aplicaciones. El controlador se encarga de aislar al modelo y a la vista de los detalles del protocolo utilizado para las peticiones. El modelo se encarga de la

abstracción de la lógica relacionada con los datos, haciendo que la vista y las acciones sean independientes de, por ejemplo, el tipo de gestor de bases de datos utilizado por la aplicación. El uso de un framework que utiliza MVC obliga a dividir y organizar el código de acuerdo a las convenciones establecidas por el mismo. El código de la presentación se guarda en la vista, el código de manipulación de datos se guarda en el modelo y la lógica de procesamiento de las peticiones constituye el controlador.

1.3.3 Sistemas Gestores de Base de Datos.

MySQL

MySQL es un sistema de gestión de bases de datos relacional, licenciado bajo la GPL de la GNU. Su diseño multihilo le permite soportar una gran carga de forma muy eficiente. MySQL fue creada por la empresa sueca MySQL AB, que mantiene el copyright del código fuente del servidor SQL, así como también de la marca.

Características:

- Gran portabilidad entre sistemas.
- Soporta hasta 32 índices por tabla.
- Soporta gran cantidad de tipos de datos para las columnas.
- Dispone de API's en gran cantidad de lenguajes (C, C++, Java, PHP).
- Aprovecha la potencia de sistemas multiprocesador, gracias a su implementación multihilo.
- Gestión de usuarios y passwords, manteniendo un muy buen nivel de seguridad en los datos.

Ventajas:

- Gran velocidad a la hora de realizar las operaciones, lo que le hace uno de los gestores que ofrecen mayor rendimiento.
- Su bajo consumo lo hacen apto para ser ejecutado en una máquina con escasos recursos sin ningún problema.
- Las utilidades de administración de este gestor son envidiables para muchos de los gestores comerciales existentes, debido a su gran facilidad de configuración e instalación.
- Tiene una probabilidad muy reducida de corromper los datos, incluso en los casos en los que los errores no se produzcan en el propio gestor, sino en el sistema en el que está.
- El conjunto de aplicaciones Apache-PHP-MySQL es uno de los más utilizados en Internet en servicios de foro (Barrapunto.com) y de buscadores de aplicaciones (Freshmeat.net).

Desventajas:

- Carece de soporte para transacciones, rollback's y subconsultas.

Capítulo 1: Fundamentación Teórica.

- No es viable para su uso con grandes bases de datos, a las que se acceda continuamente, ya que no implementa una buena escalabilidad.
- El hecho de que no maneje la integridad referencial, hace de este gestor una solución pobre para muchos campos de aplicación, sobre todo para aquellos programadores que provienen de otros gestores que sí poseen esta característica.

PostgreSQL.

PostgreSQL es un sistema gestor de base de datos objeto-relacional (*Object Relational Database Manager System*), basado en POSTGRES Versión 8.2, desarrollado en la Universidad de California, en el Departamento de Ciencias de la Computación de Berkeley.

Postgres es pionero en muchos conceptos que solo estuvieron disponibles en algunos sistemas de bases de datos comerciales mucho más tarde. PostgreSQL es un descendiente del “código abierto”.

Soporta gran parte del SQL estándar y muchas modernas funcionalidades como:

- Consultas complejas.
- Llaves foráneas.
- Disparadores.
- Vistas.
- Integridad transaccional.
- Control de versionado concurrente. Estrategia de almacenamiento que permite trabajar con grandes volúmenes de datos.

Cumple completamente con las características ACID (acrónimo de Atomicity, Consistency, Isolation and Durability: Atomicidad, Consistencia, Aislamiento y Durabilidad en español) para realizar transacciones seguras, es multiplataforma, está disponible para 34 plataformas en su última versión estable. Posee interfaces nativas para lenguajes como ODBC, JDBC, C, C++, PHP, PERL, TCL, ECPG; PYTHON y RUBY, además de traer soporte para la herencia y la seguridad de la capa de dispositivo de transportación de datos.

Además, PostgreSQL puede ser personalizado por el usuario en muchas formas, según sus necesidades, por ejemplo, adicionando entre otros, un nuevo:

- Tipo de datos.
- Funciones.

- Operadores.
- Funciones agregadas.
- Lenguajes procedurales.

Y debido a la liberación de la licencia, PostgreSQL puede ser utilizado, modificado y distribuido por cualquiera gratuitamente, para cualquier propósito ya sea con fines privados, comerciales o académicos. Por todo lo antes mencionado es que PostgreSQL es el servidor de base de datos seleccionado.

1.3.4 Metodologías de desarrollo de software: Rational Unified Process (RUP).

Rational Unified Process (RUP) es un proceso de desarrollo de software y junto con el Lenguaje Unificado de Modelado (UML), constituye la metodología estándar más utilizada para el análisis, implementación y documentación de sistemas orientados a objetos.

RUP no es un sistema con pasos firmemente establecidos, sino un conjunto de metodologías adaptables al contexto y necesidades de cada organización, utiliza UML para definir los modelos de software y se divide en 4 fases: Inicio, Elaboración, Construcción y Transición.

- Inicio: Tiene como objetivo determinar la visión del proyecto.
- Elaboración: En esta etapa el objetivo es determinar la arquitectura óptima.
- Construcción: El objetivo es obtener la capacidad operacional inicial.
- Transición: Obtener el reléase del producto.

En RUP se han agrupado las actividades en grupos lógicos definiéndose 9 flujos de trabajo principales, los 6 primeros son conocidos como flujos de ingeniería y los tres últimos como flujos de apoyo. Tiene tres características esenciales: dirigido por casos de uso, centrado en la arquitectura e iterativo e incremental.

1.3.5 Lenguaje de Modelado: UML

Lenguaje Unificado de Modelado es un lenguaje gráfico para especificar, construir, visualizar, documentar las partes o artefactos que son información utilizada y originada mediante un proceso de software. Es un lenguaje estándar de modelado orientado a objetos.

UML no puede compararse con la programación estructurada, pues no es programación, solo se diagrama la realidad de una utilización en un requerimiento. Mientras que, programación estructurada, es una forma

de programar como lo es la orientación a objetos, sin embargo, la orientación a objetos viene siendo un complemento perfecto de UML, pero no por eso se toma UML sólo para lenguajes orientados a objetos.

Los principales beneficios de UML son:

- Modelar sistemas utilizando conceptos orientados a objetos.
- Establecer conceptos y artefactos ejecutables.
- Encaminar el desarrollo del escalamiento en sistemas complejos de misión crítica.
- Crear un lenguaje de modelado utilizado tanto por humanos como por máquinas.
- Mejor soporte a la planeación y al control de proyectos.
- Alta reutilización y minimización de costos.

Es importante resaltar que UML es un lenguaje para especificar y no para describir métodos o procesos. Se utiliza para definir un sistema de software, detallar los artefactos en el sistema, documentar y construir. Es el lenguaje en el que está descrito el modelo. Se puede aplicar en una gran variedad de formas para dar soporte a una metodología de desarrollo de software tal como RUP, pero no especifica en sí mismo qué metodología o proceso usar.

1.3.6 Herramientas Case: Visual Paradigm

Visual Paradigm para UML es una herramienta UML profesional que soporta el ciclo de vida completo del desarrollo de software: análisis y diseño orientados a objetos, construcción, pruebas y despliegue. El software de modelado UML ayuda a una más rápida construcción de aplicaciones de calidad, mejores y a un menor coste. Permite dibujar todos los tipos de diagramas de clases, código inverso, generar código desde diagramas y generar documentación.

Las Herramientas CASE (Computer Aided Software Engineering, Ingeniería de Software Asistida por Ordenador) son diversas aplicaciones informáticas destinadas a aumentar la productividad en el desarrollo de software reduciendo el coste de las mismas en términos de tiempo y de dinero. Estas herramientas ayudan en todos los aspectos del ciclo de vida de desarrollo del software en tareas como el proceso de realizar un diseño del proyecto, cálculo de costes, implementación de parte del código automáticamente con el diseño dado, compilación automática, documentación o detección de errores.

Visual Paradigm es una poderosa herramienta para visualizar y diseñar elementos de software, para ello utiliza UML y ofrece una gama de facilidades para el modelado de aplicaciones. Está orientada a la creación de diseños usando el paradigma de programación orientada a objetos.

Tiene dentro de sus características que es portable y posee gran facilidad de uso. Utiliza UML como lenguaje de modelado. Su diseño se centra en casos de uso y se enfoca al negocio que genera un software de mayor calidad. También tiene disponibilidad en múltiples plataformas y usa un lenguaje estándar común a todo el equipo de desarrollo que facilita la comunicación.

Se decide utilizar Visual Paradigm porque es una herramienta UML profesional que soporta el ciclo de vida completo del desarrollo de software: análisis y diseño orientados a objetos, construcción, pruebas y despliegue. Permite dibujar todos los tipos de diagramas de clases, código inverso, generar código desde diagramas y generar documentación.

1.4 Conclusiones.

En este capítulo se han introducido conceptos indispensables para la comprensión del proceso de distribución de la información, con el uso de las tecnologías actuales. La mejor solución al problema de gestión de los cursos optativos en la Facultad 1 es la implementación de un sistema basado en web. Se hace necesaria la construcción de un software que ayude controlar el proceso de gestión de los CO en la Facultad 1 de la Universidad de las Ciencias Informáticas. También se detallaron las condiciones y problemas que rodean al objeto de estudio; y a través de los conceptos y definiciones planteadas, se determinaron las condiciones específicas que rodean al problema y en base a esto se obtuvieron los objetivos generales y específicos para este trabajo.

PROPUESTA 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 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. 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 y aprobados cierto número de cursos optativos.

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, se envía el que lo impartirá, la capacidad 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, todo depende de la necesidad de la Facultad y la estrategia que se tome en este sentido. Una vez vencido el plazo, se conforman los grupos de cada curso y se le entrega o envía por vía electrónica al que lo impartirá, quien se encargaría de informar a sus estudiantes sobre todo lo referente al curso hasta su culminación.

Capítulo 2: Descripción de la Solución Propuesta.

El negocio termina cuando vence la fecha límite de la matrícula y el profesor que impartirá el curso recibe el listado de los estudiantes matriculados.

2.2.1. 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 1 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.

Tabla 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.
Cliente	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.

Tabla 2: Descripción de los Trabajadores del Negocio.

Trabajadores del negocio	Justificación
Secretaria docente	Encargada de matricular a los estudiantes, brindar cualquier información referente a los cursos o el proceso de matrícula, que le sea solicitada.

Capítulo 2: Descripción de la Solución Propuesta.

Profesor	Encargado de impartir el curso.
----------	---------------------------------

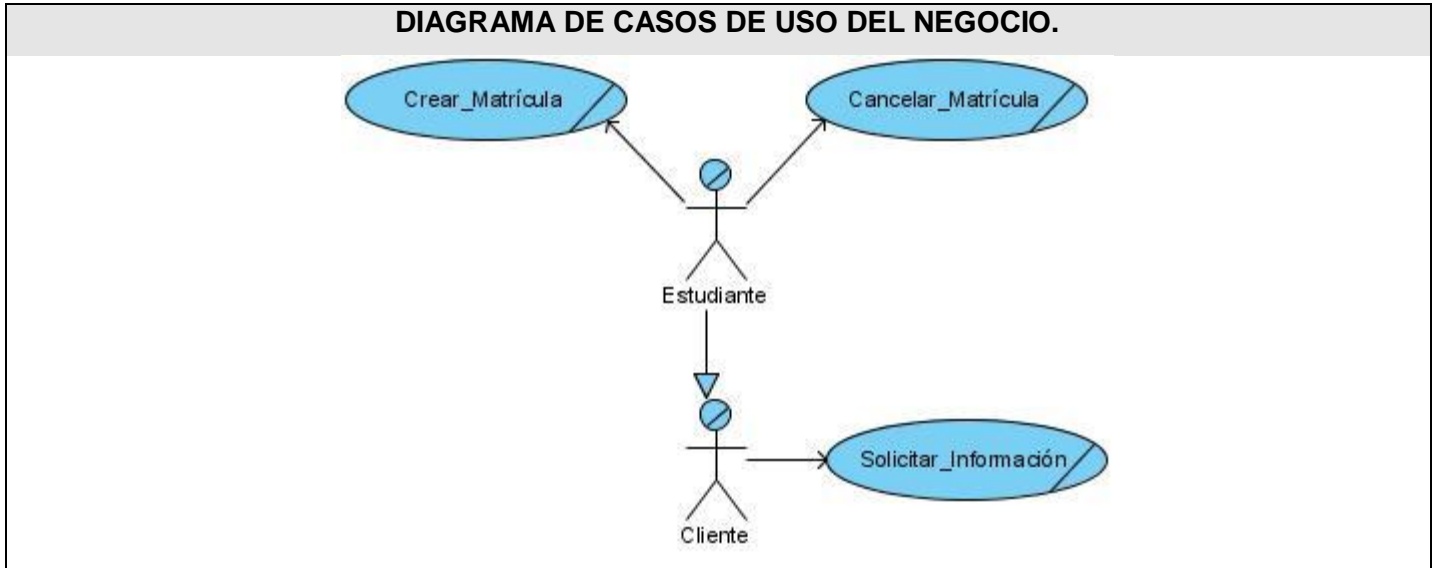


Figura 1: Diagrama de Casos de Uso del Negocio.

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

Descripción de los Casos de Uso del Negocio.

Tabla 3: Descripción textual del CUN Crear Matrícula.

Caso de Uso del Negocio	Crear_Matrícula
Actores	Estudiante
Propósito	Que los estudiantes puedan matricularse en un

Capítulo 2: Descripción de la Solución Propuesta.

	curso optativo según su interés o necesidad, como resultado de la planificación y organización de estos por parte del responsable.
Casos de uso Asociados	Ninguno
Resumen <p>El caso de uso se inicia cuando el estudiante, ante la petición por parte de la Facultad o en su interés propio y luego de la convocatoria realizada por correo electrónico u otra vía por el Decano, se presenta en la Secretaría a solicitar matricularse en uno o varios cursos optativos de los que se ofertan. Esta solicitud será atendida por la secretaria docente que evalúa si puede o no matricularlo en dependencia de su año o la capacidad del CO solicitado.</p>	
Flujo de Trabajo	
Acciones del actor	Respuesta del proceso de negocio
1. El estudiante se presenta en la Secretaría para solicitar matricularse en uno o varios cursos según necesidades de la Facultad o interés personal.	2. La secretaria le pregunta qué curso desea y pide los datos: nombre y solapín.
3. El estudiante aporta los datos pedidos.	4. La secretaria incluye a este estudiante en la lista del curso que ha elegido, de no existir inconveniente.
Cursos alternos:	
Acciones del actor	Respuesta del proceso de negocio
	4.1 Si el curso optativo solicitado presenta su capacidad cubierta, la secretaria deniega la solicitud.
Prioridad	Crítica

Capítulo 2: Descripción de la Solución Propuesta.

Tabla 4: Descripción textual del CUN Cancelar Matrícula.

Caso de Uso del Negocio	Cancelar_Matrícula
Actores	Estudiante
Propósito	Que los estudiantes puedan cancelar una matrícula en un curso optativo según su interés personal.
Casos de uso Asociados	Ninguno
Resumen	
El caso de uso se inicia cuando el estudiante, por interés propio, se presenta en la Secretaría a solicitar cancelar una o varias matrículas realizadas. Esta solicitud será atendida por la secretaria docente que evalúa si el estudiante se encontraba matriculado y si lo estaba cancela la matrícula y actualiza la capacidad del curso.	
Flujo de Trabajo	
Acciones del actor	Respuesta del proceso de negocio
1. El estudiante se presenta en la Secretaría para solicitar cancelar una o varias matrículas realizadas según su interés personal.	2. La secretaria le pregunta qué curso desea cancelar y pide los datos: nombre y solapín.
3. El estudiante aporta los datos pedidos.	4. La secretaria verifica que la matrícula se hubo realizado. 5. Si todo es correcto la secretaria cancela la(s) matrículas solicitadas y actualiza la capacidad del curso.
Cursos alternos:	
Acciones del actor	Respuesta del proceso de negocio

Capítulo 2: Descripción de la Solución Propuesta.

	5.1 Si la matrícula no se hubo realizado la secretaria informa al estudiante.
Prioridad	Crítica

Tabla 5: Descripción textual del CUN Solicitar Información.

Caso de Uso del Negocio	Solicitar_información
Actores del Negocio	Cliente
Propósito	El propósito es que el cliente pueda pedir información al negocio.
Casos de uso Asociados	Ninguno
Resumen El caso de uso se inicia cuando algún cliente de la Facultad solicita alguna información, estadística o reporte sobre los cursos optativos. 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.	
Flujo de Trabajo	
Acciones del actor	Respuesta del proceso de negocio
1 El Cliente solicita una información, reporte o estadística del estado de los cursos optativos.	2 Los trabajadores del negocio procesan la información que poseen según las necesidades del cliente.
3 El Cliente recibe la información, reporte o estadística.	
Prioridad	Crítica
Cursos alternos: -----	

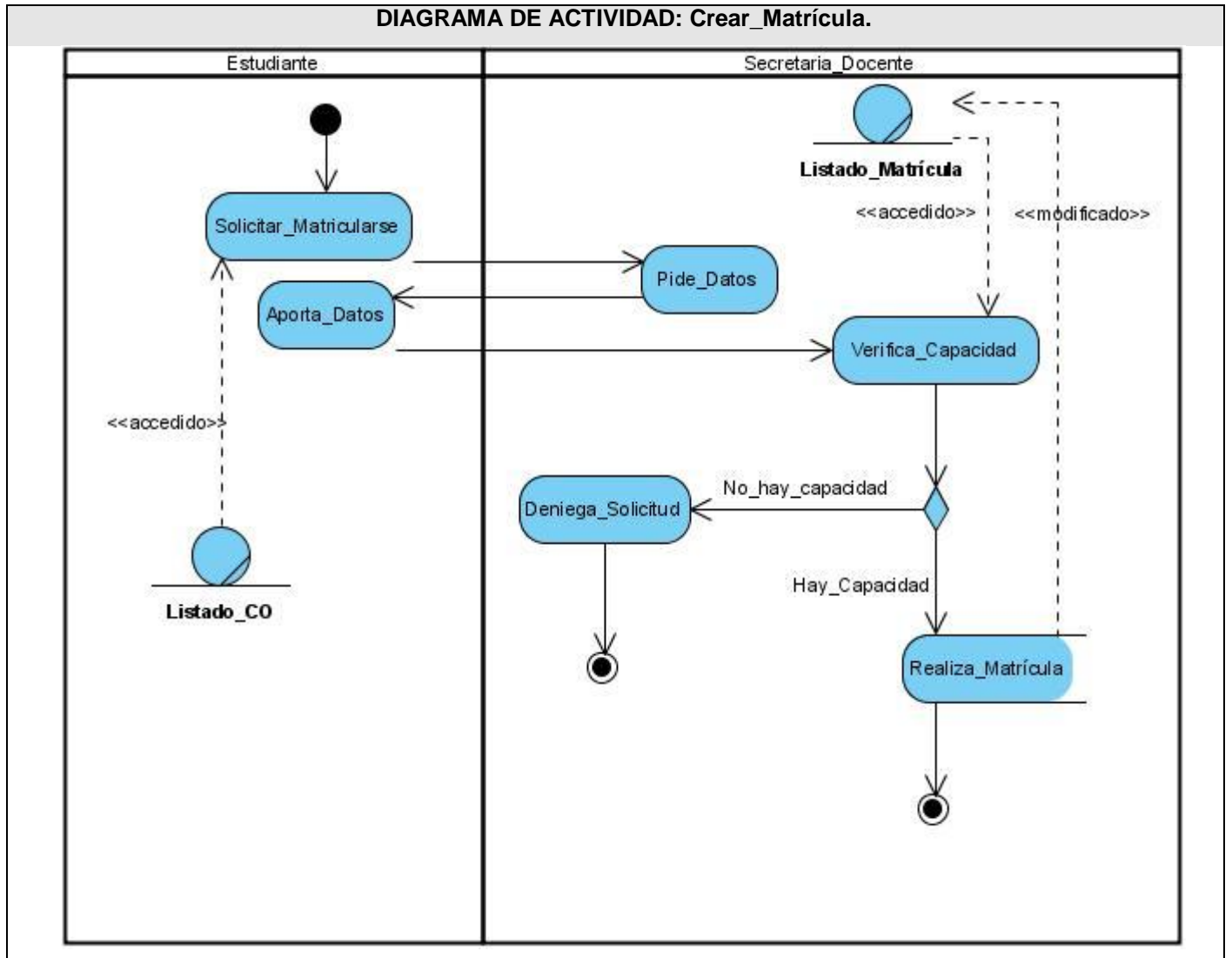


Figura 2: Diagrama de Actividades Crear Matrícula.

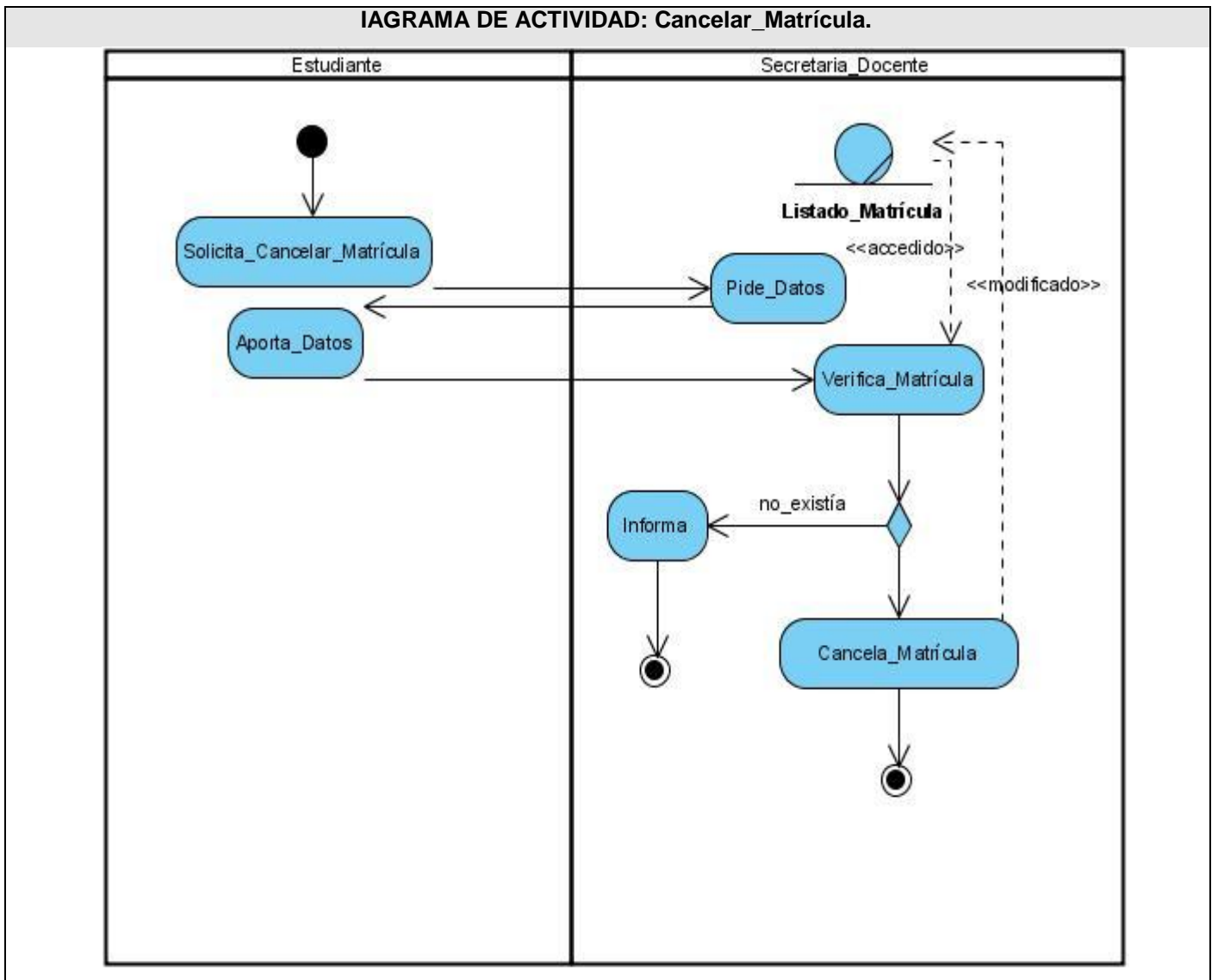


Figura 3: Diagrama de Actividades Cancelar Matrícula.

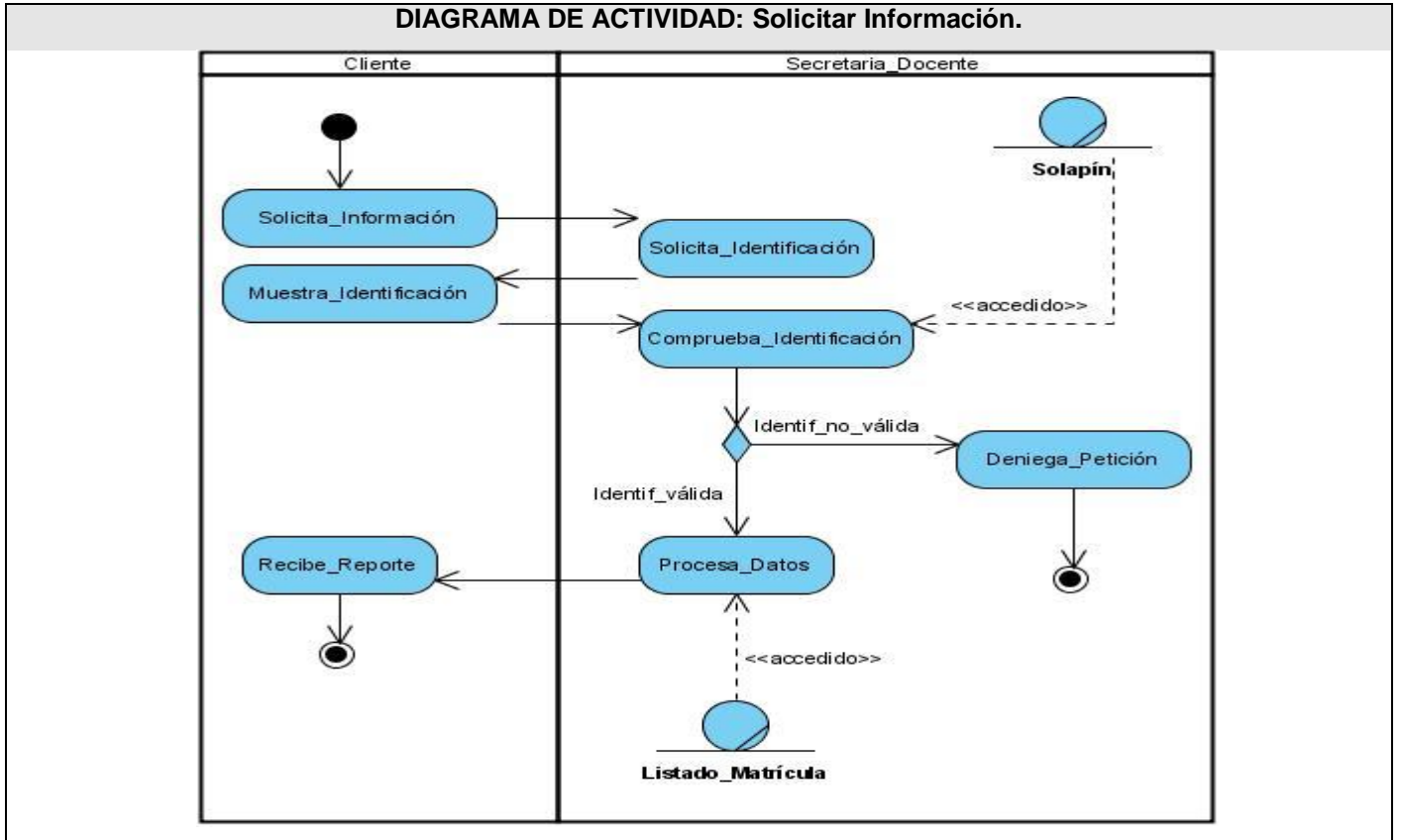


Figura 4: Diagrama de Actividades Solicitar Información.

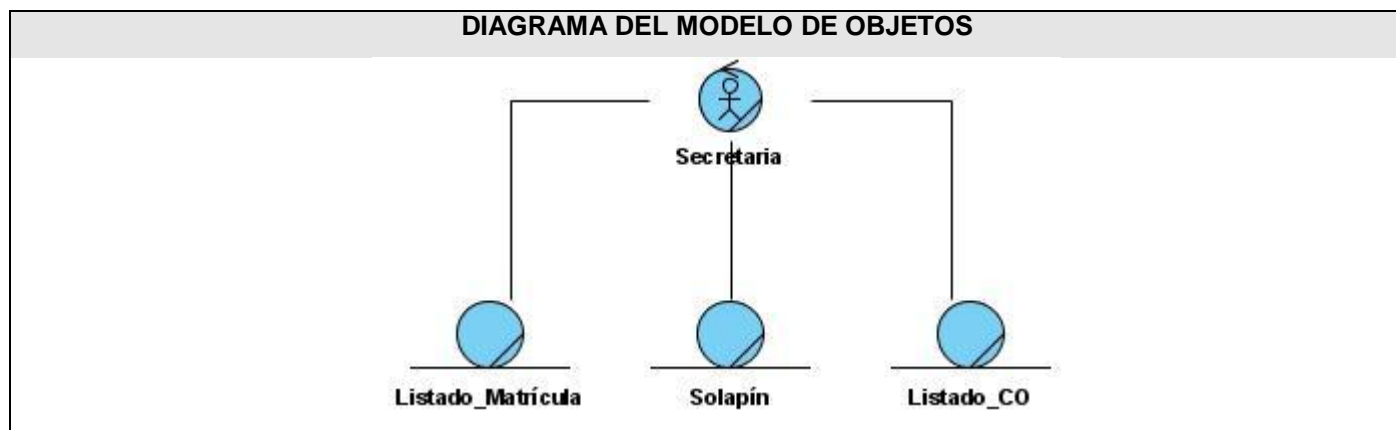


Figura 5: Diagrama del Modelo de Objetos.

2.3. Requerimientos funcionales.

RF 1 Autenticar Usuario.

1. El sistema debe mostrar un formulario a llenar, solicitando usuario y contraseña.
2. El sistema debe validar los datos entrados.

RF 2 Gestionar curso optativo.

1. Crear Curso Optativo.
 - 1.1. El sistema debe mostrar un formulario a llenar, dándole la posibilidad a los profesores de crear un curso a partir de los parámetros establecidos con tal fin.
 - 1.2. El sistema debe almacenar la información del curso (tema del nuevo curso optativo, solapín del profesor, perfil, disciplina, fecha de comienzo, capacidad de matrícula, cantidad de horas clase).
 - 1.3. El sistema debe notificar que se creó el curso satisfactoriamente y finalizar el caso de uso.
2. Modificar curso optativo.
 - 2.1. El sistema debe mostrar un formulario con todos los cursos creados.
 - 2.2. El sistema debe permitir seleccionar el curso a modificar.
 - 2.3. El sistema debe mostrar un formulario con los parámetros modificables del curso.
 - 2.3.1. Si se modifica algún parámetro del curso, el sistema debe notificarlo y almacenar los cambios realizados.
3. Eliminar curso optativo.
 - 3.1. El sistema debe mostrar un formulario con todos los cursos creados.
 - 3.2. El sistema debe permitir seleccionar el curso a eliminar.

Capítulo 2: Descripción de la Solución Propuesta.

3.3. El sistema debe mostrar un mensaje de confirmación.

3.3.1. Si se ratifica la opción de eliminar el sistema debe mostrar nuevamente el listado de los cursos.

4. Habilitar curso optativo.

4.1. El sistema debe mostrar el listado de los cursos no habilitados.

4.2. El sistema debe permitir seleccionar el curso que se desea habilitar.

4.3. Una vez habilitado un curso, el sistema deberá mostrar nuevamente el listado de los cursos no habilitados para confirmar que ha sido habilitado correctamente.

RF 3 Gestionar Matrícula.

1. Crear Matrícula.

1.1. El sistema debe mostrar el listado de los cursos habilitados.

1.2. El sistema debe permitir seleccionar un curso dentro de los habilitados.

1.2.1. El sistema debe verificar que el estudiante no se ha matriculado previamente.

1.2.2. El sistema debe brindar la posibilidad de regresar al menú anterior.

1.3. El sistema debe almacenar la información inherente a la matrícula (curso optativo y solapín del estudiante).

1.4. El sistema debe notificar que se realizó la matrícula satisfactoriamente y finalizar el caso de uso.

2. Cancelar Matrícula.

2.1. El sistema debe mostrar el listado de los cursos en los que el estudiante se ha matriculado.

2.2. El sistema debe permitir seleccionar un curso a eliminarle la matrícula.

2.3. El sistema debe volver a mostrar el listado de los cursos en los que se ha matriculado, notificando q ha sido eliminada su anterior matricula satisfactoriamente.

3. Crear Matrícula Forzada.

3.1. El sistema debe mostrar el listado de los cursos habilitados.

3.2. El sistema debe permitir seleccionar un curso dentro de los habilitados.

3.3. El sistema debe mostrar un formulario con los campos a llenar.

3.3.1. El sistema debe verificar que el estudiante no se ha matriculado previamente.

3.4. El sistema debe mostrar un mensaje de confirmación y regresar al listado de los cursos habilitados.

Capítulo 2: Descripción de la Solución Propuesta.

4. Cancelar Matrícula Forzada.

4.1. El sistema debe mostrar el listado de los cursos habilitados.

4.2. El sistema debe permitir seleccionar un curso dentro de los habilitados.

4.3. El sistema debe mostrar el listado de los estudiantes matriculados en dicho curso.

4.4. El sistema debe mostrar un mensaje de confirmación y regresar al listado de los cursos habilitados.

RF 4 Gestionar información.

1. Gestionar Rol.

1.1. Crear Rol.

1.1.1. El sistema debe mostrar un formulario con los campos a llenar.

1.1.2. El sistema debe permitir asignar funcionalidades al rol a crear.

1.2. Modificar Rol.

1.2.1. El sistema debe mostrar el listado de los roles existentes y brindar la posibilidad de seleccionar uno.

1.2.2. El sistema debe mostrar un formulario con los datos a modificar que tiene que incluir la funcionalidad de dicho rol.

1.2.3. El sistema debe almacenar los parámetros modificados y finalizar el caso de uso.

1.3. Eliminar Rol.

1.3.1. El sistema debe mostrar el listado de los roles existentes.

1.3.2. El sistema debe permitir seleccionar el rol a eliminar y mostrar un mensaje de confirmación.

1.3.3. El sistema debe eliminar el rol y finalizar el caso de uso.

2. Gestionar Perfil.

2.1. Crear Perfil.

2.1.1. El sistema debe mostrar un formulario con los campos a llenar.

2.1.2. El sistema debe validar, almacenar los parámetros entrados y finalizar el caso de uso.

2.2. Modificar Perfil.

2.2.1. El sistema debe mostrar el listado de los perfiles existentes y permitir seleccionar uno.

2.2.2. El sistema debe mostrar un formulario con los parámetros a modificar.

2.2.3. El sistema debe validar los parámetros entrados, almacenarlos y finalizar el caso de uso.

2.3. Eliminar Perfil

Capítulo 2: Descripción de la Solución Propuesta.

2.3.1. El sistema debe mostrar el listado de los perfiles existentes.

2.3.2. El sistema debe permitir seleccionar el perfil a eliminar y mostrar un mensaje de confirmación.

2.3.3. El sistema debe eliminar el perfil y finalizar el caso de uso.

3. Gestionar Disciplina.

3.1. Crear Disciplina.

3.1.1. El sistema debe mostrar un formulario con los campos a llenar.

3.1.2. El sistema debe validar, almacenar los parámetros entrados y finalizar el caso de uso.

3.2. Modificar Disciplina.

3.2.1. El sistema debe mostrar el listado de las disciplinas existentes y permitir seleccionar una.

3.2.2. El sistema debe mostrar un formulario con los parámetros a modificar.

3.2.3. El sistema debe validar los parámetros entrados, almacenarlos y finalizar el caso de uso.

3.3. Eliminar Disciplina

3.3.1. El sistema debe mostrar el listado de las Disciplinas existentes.

3.3.2. El sistema debe permitir seleccionar la Disciplina a eliminar y mostrar un mensaje de confirmación.

3.3.3. El sistema debe eliminar la Disciplina y finalizar el caso de uso.

2.4. Requisitos No Funcionales.

Los requisitos no funcionales son propiedades o cualidades que el producto debe tener, o sea, características que hagan al producto atractivo, rápido, usable o confiable. Están estrechamente vinculados a los requisitos funcionales, puesto que una vez que esta definido lo que el sistema debe hacer, es necesario especificar cómo ha de hacerlo. Pueden llegar a marcar la diferencia entre un producto bien aceptado por los clientes y usuarios y uno de poca o ninguna calidad y aceptación.

- Interfaz externa

Diseño orientado a llamar la atención del usuario y con una navegación sencilla. Diseño intuitivo que permita a los usuarios usar cómoda y fácilmente la aplicación.

- Usabilidad

La aplicación está concebida para un número limitado de usuarios, sólo podrá ser utilizado por aquellas personas que estén autorizadas.

- Rendimiento

La aplicación tiene un alto nivel de respuesta, tanto para los accesos a la BD, como para los diferentes procesos que realiza.

- Portabilidad

La aplicación es multiplataforma (Windows/Linux).

- Seguridad

La aplicación está concebida para brindar los permisos necesarios a cada uno de los usuarios que tienen autorización a utilizarlas según los roles de los mismos.

- Software

Está programado en el lenguaje PHP, utiliza como framework el Symfony 1.0.18 y como IDE de desarrollo el Eclipse.

- Confiabilidad

Tiene un alto grado de confiabilidad, dado su fin de gestionar la información.

- Ayuda y documentación en línea.

Cuenta con un manual de usuario, que permite la aclaración de cualquier duda al interactuar con la aplicación.

- Administración

Fácil administración pues está bien organizado por paquetes de programación. Alguna modificación en la programación debe hacerse en el paquete correspondiente sin afectar a los demás.

2.5. Definición de los Procesos del sistema.

2.5.1. Descripción del sistema propuesto.

El sistema a desarrollar constará de una interfaz sencilla y fácil de usar, diseñada para usuarios no adiestrados en el campo de la informática, con el objetivo de simplificar la interacción de los mismos con el sistema, este se implementará en PHP y se diseñará en forma de ventanas, o sea mediante elementos desplegables por donde se puedan seleccionar las opciones deseadas sin necesidad de la entrada excesiva de datos. Se realizará además un mapeo de roles para asignar a cada rol participante únicamente las funcionalidades que le corresponden evitando así pérdidas de información e inseguridad de la misma, respetando siempre la disponibilidad de la información que a cada rol concierne, por ejemplo:

Capítulo 2: Descripción de la Solución Propuesta.

los estudiantes solo tendrán el acceso necesario a la información necesaria para matricularse o darse baja de un cursos optativo, y solo tendrán visibles aquellos cursos optativos que hayan sido previamente aprobados y habilitados.

Definición de los actores del sistema.

Tabla 6: Descripción de los Actores del Sistema.

Actores del sistema:	
<u>Actor</u>	<u>Justificación</u>
Administrador	Persona especializada en el trabajo con la aplicación, es el que va a gestionar algunos de los parámetros que formarán parte de los CO como son los roles, perfiles, años y las disciplinas, así como habilitar los CO cuando se oriente y gestionar las matrículas forzadas.
Profesor	Es el encargado de introducir los parámetros de los CO, para la creación de los mismos.
Estudiante	Es el encargado de matricularse en los CO que se estén ofertando así como cancelar una matrícula ya realizada.

Casos de uso.

Tabla 7: Paquetes de CUS.

Cód.	Nombre de caso de uso	Paquete	Justificación de la selección.
CU-1	Autenticarse		
CU-2	Gestionar Matrícula		
CU-3	Gestionar Perfil		
CU-4	Gestionar Disciplina		
CU-5	Gestionar Año		
CU-6	Gestionar Administrador		

Capítulo 2: Descripción de la Solución Propuesta.

CU-7	Gestionar Matrícula Forzada		
CU-8	Gestionar CO		

Diagrama de Casos de Uso.

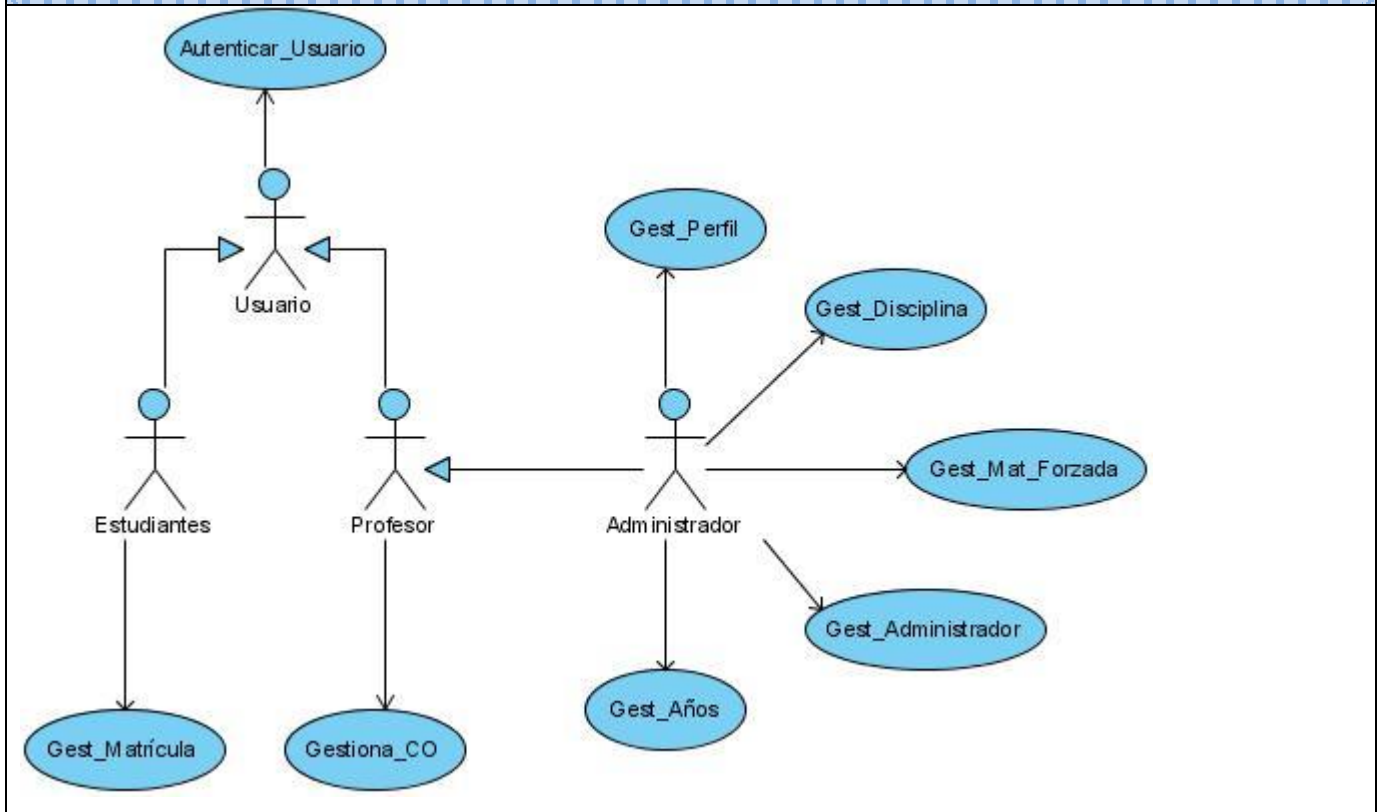


Figura 6: Diagrama de Casos de Uso del Sistema.

Descripción de los Casos de Uso del Sistema

Tabla 8: Descripción textual del CUS Autenticar Usuario.

CU-1	Autenticar_Usuario
Actores	Usuario (Profesor, Estudiante y Administrador)
Propósito: Permitir a los profesores, estudiantes y administrador acceder a las funcionalidades del sistema.	

Capítulo 2: Descripción de la Solución Propuesta.

Resumen: El caso de uso permite que los usuarios con determinadas responsabilidades en el sistema introduzcan sus credenciales (usuario y contraseña) para que el sistema las verifique y ejecuten las funcionalidades que puedan según su rol.

Referencias: RF1

Interfaz I

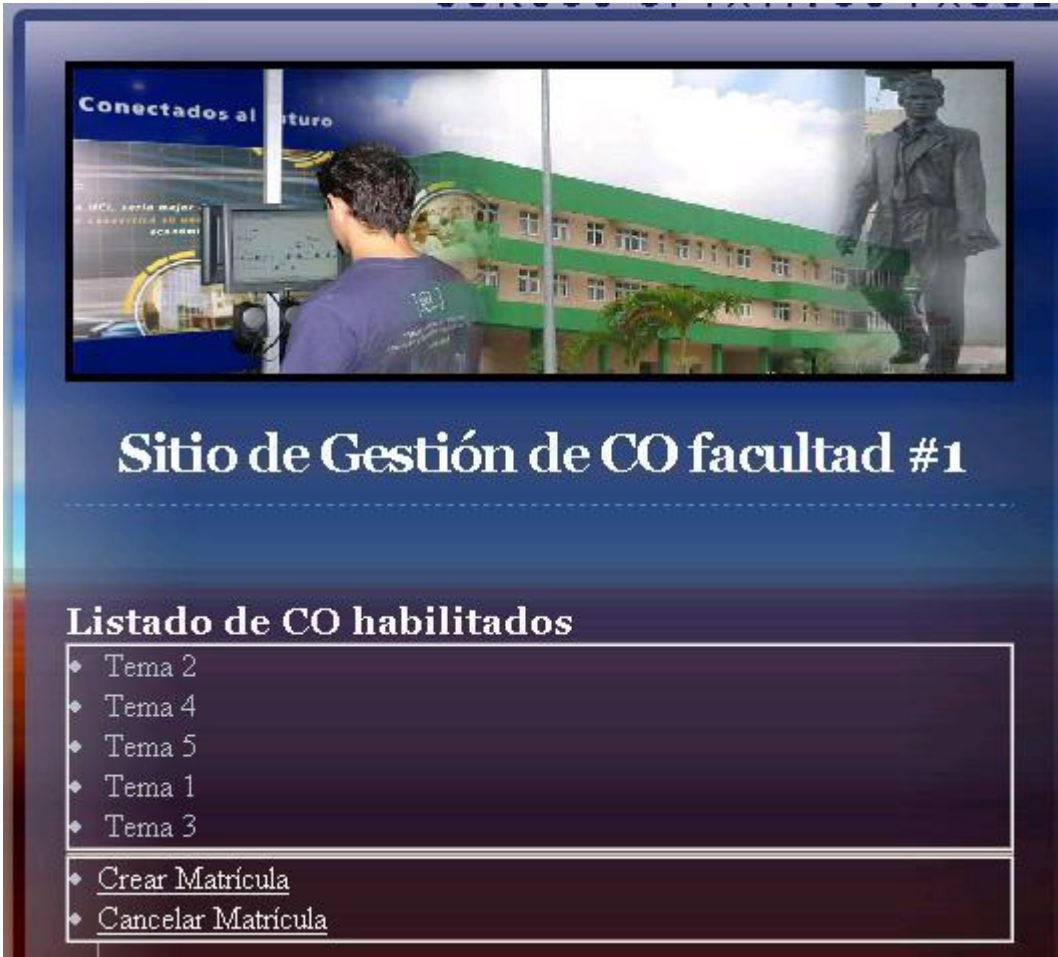


Flujo Normal de los Eventos

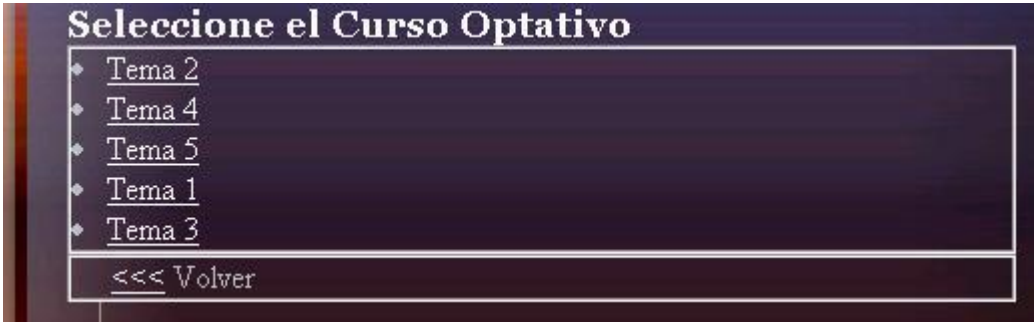
Acción del actor	Respuesta del sistema
1. El usuario introduce sus datos en el sistema (usuario y contraseña).	2. Se validan los datos entrados. 3. Si los datos son correctos se almacenan. 4. Si los datos son incorrectos se muestra un mensaje de error y se le permite introducir los datos nuevamente.

Capítulo 2: Descripción de la Solución Propuesta.

Tabla 9: Descripción textual del CUS Gestionar Matrícula.

CU-2	Gestionar Matrícula
Actores	Estudiante
Propósito: Que los estudiantes puedan acceder a los cursos optativos habilitados con el fin de realizar las matrículas deseadas.	
Resumen: El caso de uso se inicia cuando algún estudiante desea matricularse en un determinado curso o eliminar una matrícula previamente realizada. El caso de uso termina cuando se realizan algunas de estas acciones.	
Referencias	
Interfaz I	
	

Capítulo 2: Descripción de la Solución Propuesta.

Flujo Normal de los Eventos	
Sección "Matricular"	
Acción del actor	Respuesta del sistema
1. Selecciona la opción Cursos Optativos Habilitados.	2. Muestra el listado de los CO habilitados.
3. Selecciona la opción matricular.	4. Realiza la matrícula.
	5. Vuelve a mostrar el listado de los cursos optativos habilitados.
Interfaz II	
	
Flujo Normal de los Eventos	
Sección "Cancelar Matricula"	
Acción del actor	Respuesta del sistema
1. Selecciona la opción Cancelar Matrícula.	2. Muestra el listado de los CO en los que se ha matriculado.
3. Selecciona el curso que desea se le cancele la matrícula.	4. Cancela la matrícula del curso seleccionado.
	5. Vuelve a mostrar el listado de los CO en los que se ha matriculado.
Interfaz III	

Capítulo 2: Descripción de la Solución Propuesta.

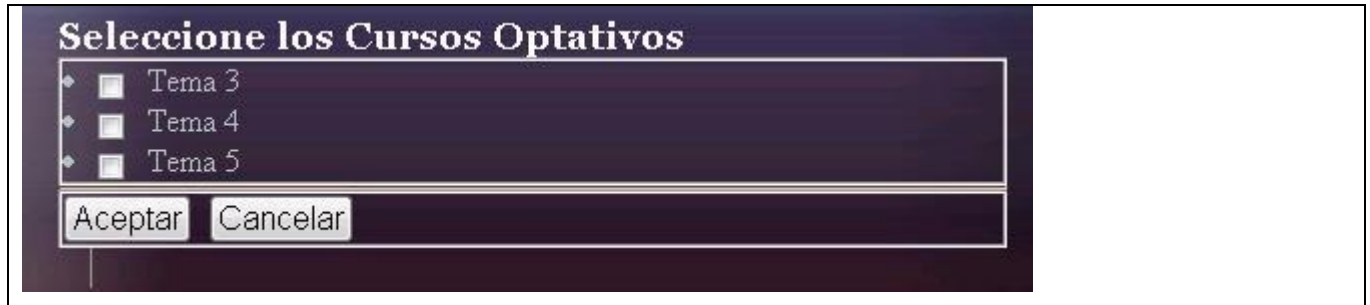
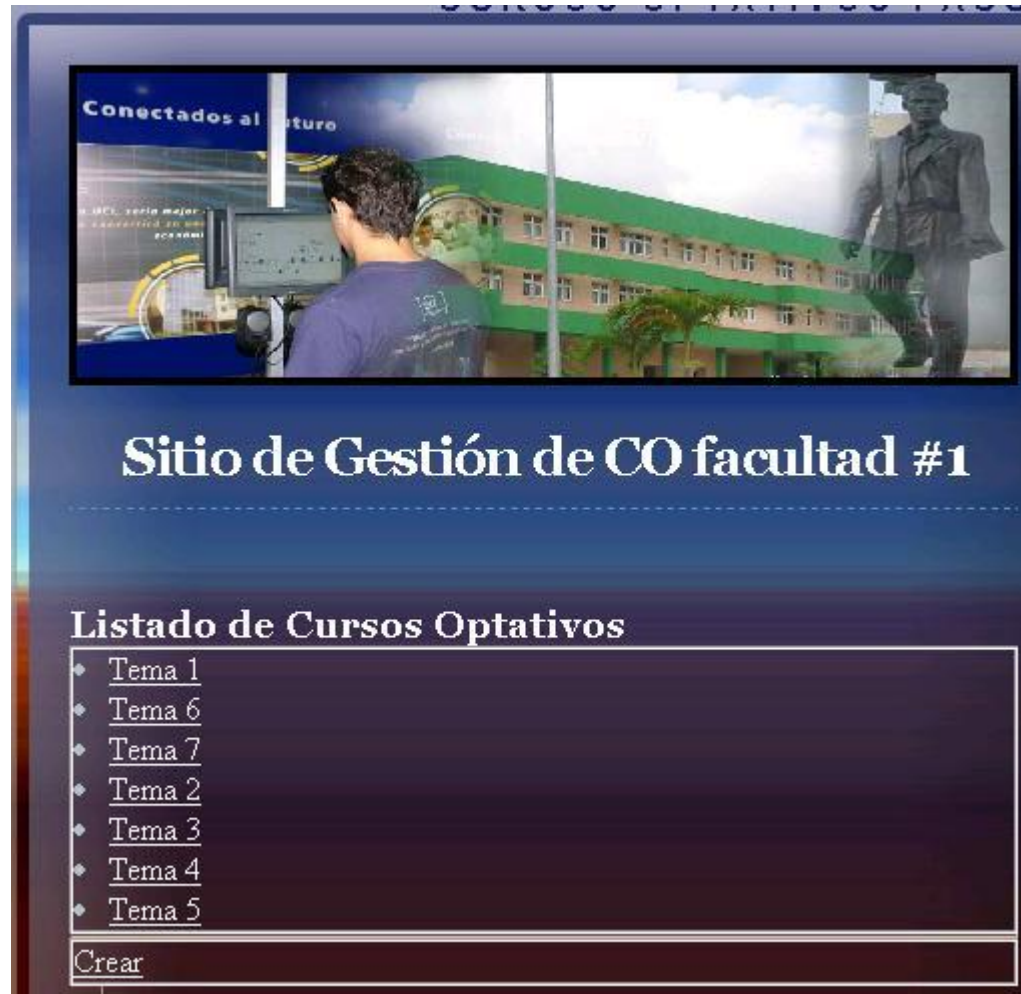


Tabla 15: Descripción textual del CUS Gestionar CO.

CU-8	Gestionar CO
Actores	Profesor
Propósito: Permitir al profesor adicionar o eliminar un CO del sistema.	
Resumen: El caso de uso se inicia cuando el profesor del sistema desea adicionar, modificar o eliminar un CO. El caso de uso termina cuando se realiza algunas de estas acciones.	
Referencias	

Interfaz



Flujo Normal de Eventos

Sección "Adicionar CO"

Acción del actor	Respuesta del sistema
1. El administrador selecciona la opción gestionar CO.	2. Se muestra el listado de los CO existentes.
3. Selecciona la opción adicionar.	4. Muestra los campos a llenar.
5. Introduce los datos en los campos. 6. Presiona el botón aceptar.	7. Se validan los datos entrados.

Flujos alternativos

Capítulo 2: Descripción de la Solución Propuesta.

Acción del actor	Respuesta del sistema
	<p>7.1. Si alguno de los datos entrados es incorrecto, se muestra un mensaje de error y se permite entrar los datos nuevamente.</p> <p>7.2. Si el CO adicionado ya existe se muestra un mensaje y el listado de los CO existentes.</p>

Interfaz I

Introduzca los datos

Tema*:

Fecha*: ...

Duración*:

Profesor*:

Disciplina*:

Modalidad*:

Perfil*:

Matrícula*:

Años*: 1 2 3 4 5

Flujo Normal de Eventos

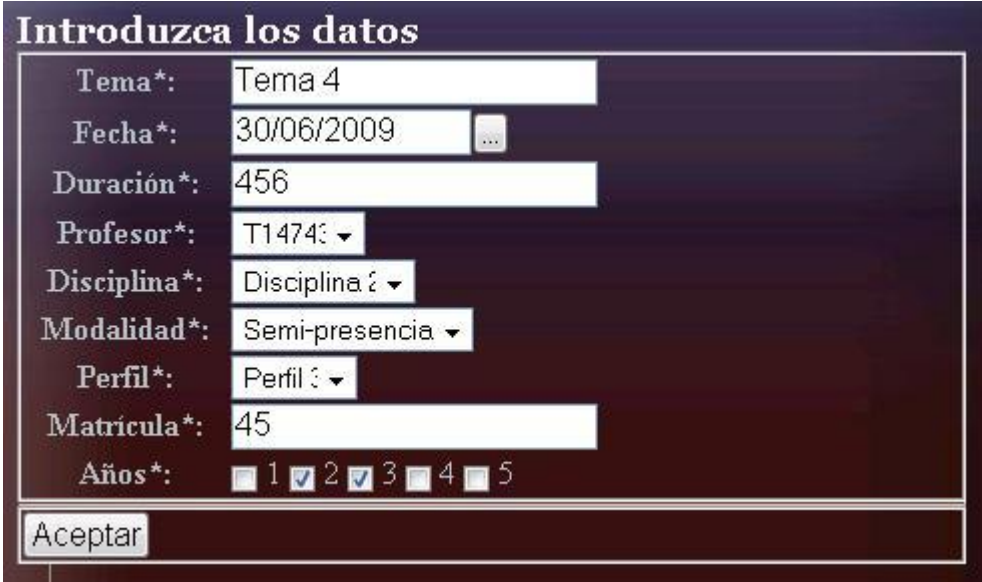
Sección "Modificar CO"

Acción del actor	Respuesta del sistema
8. El administrador selecciona la opción gestionar CO.	9. Se muestra el listado de los CO existentes.
10. Selecciona la opción modificar.	11. Muestra los datos del CO a modificar.
12. Modifica los datos en los campos. 13. Presiona el botón aceptar.	14. Se validan los datos modificados.

Flujos alternativos

Acción del actor	Respuesta del sistema

Capítulo 2: Descripción de la Solución Propuesta.

	14.1. Si alguno de los datos entrados es incorrecto, se muestra un mensaje de error y se permite entrar los datos nuevamente.
Interfaz II	
 <p>The screenshot shows a form with the following fields: Tema* (Tema 4), Fecha* (30/06/2009), Duración* (456), Profesor* (T1474), Disciplina* (Disciplina 2), Modalidad* (Semi-presencia), Perfil* (Perfil 3), Matrícula* (45), and Años* (checkboxes for 1, 2, 3, 4, 5, with 2 and 3 checked). An 'Aceptar' button is at the bottom.</p>	
Flujo Normal de Eventos	
Sección "Eliminar CO"	
Acción del actor	Respuesta del sistema
15. El administrador selecciona la opción gestionar CO.	16. Se muestra el listado de los CO existentes.
17. Selecciona la opción eliminar.	18. Muestra mensaje de confirmación.
19. Presiona el botón aceptar.	20. Se elimina el CO seleccionado.
Flujos alternativos	
Acción del actor	Respuesta del sistema
19.1. Presiona el botón cancelar.	20. Se vuelve a mostrar el listado de los CO existentes.
Interfaz III	



El resto de las descripciones se encuentran en el anexo correspondiente. **[Anexo 1]**

2.6. Conclusiones.

En este capítulo se comenzó a desarrollar la propuesta de solución, y a través de los procesos de negocio y el levantamiento de requisitos, se pudo obtener un listado con las funciones que debe tener el sistema y se describieron paso a paso todas las acciones del actor del sistema con los casos de uso con los que interactúa. A partir de esto se comenzará a desarrollar la etapa posterior que sería el análisis y diseño.

ANÁLISIS Y DISEÑO

3.1 Introducción.

La construcción de la solución propuesta tiene como base el análisis de esta a nivel de casos de uso. Aquí se representan, por cada caso de uso, los Diagramas de Clases del Análisis y Diseño, dando una vista de las clases que intervienen en el sistema.

3.2 Análisis.

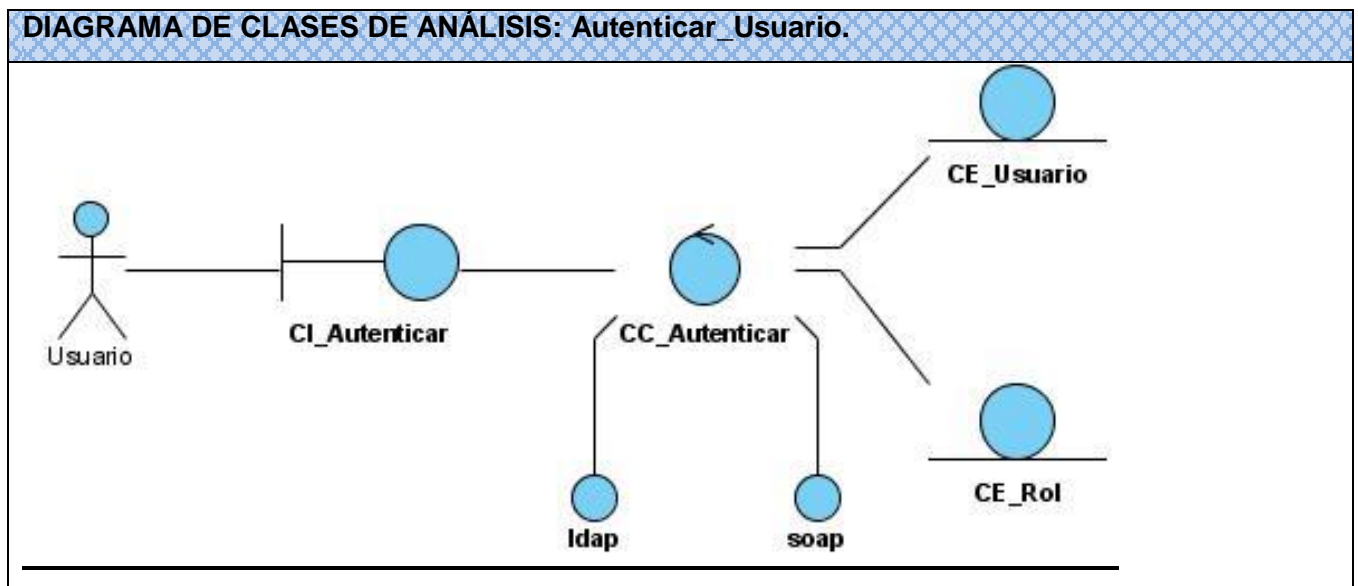


Figura 7: Diagrama de Clases del Análisis Autenticar Usuario.

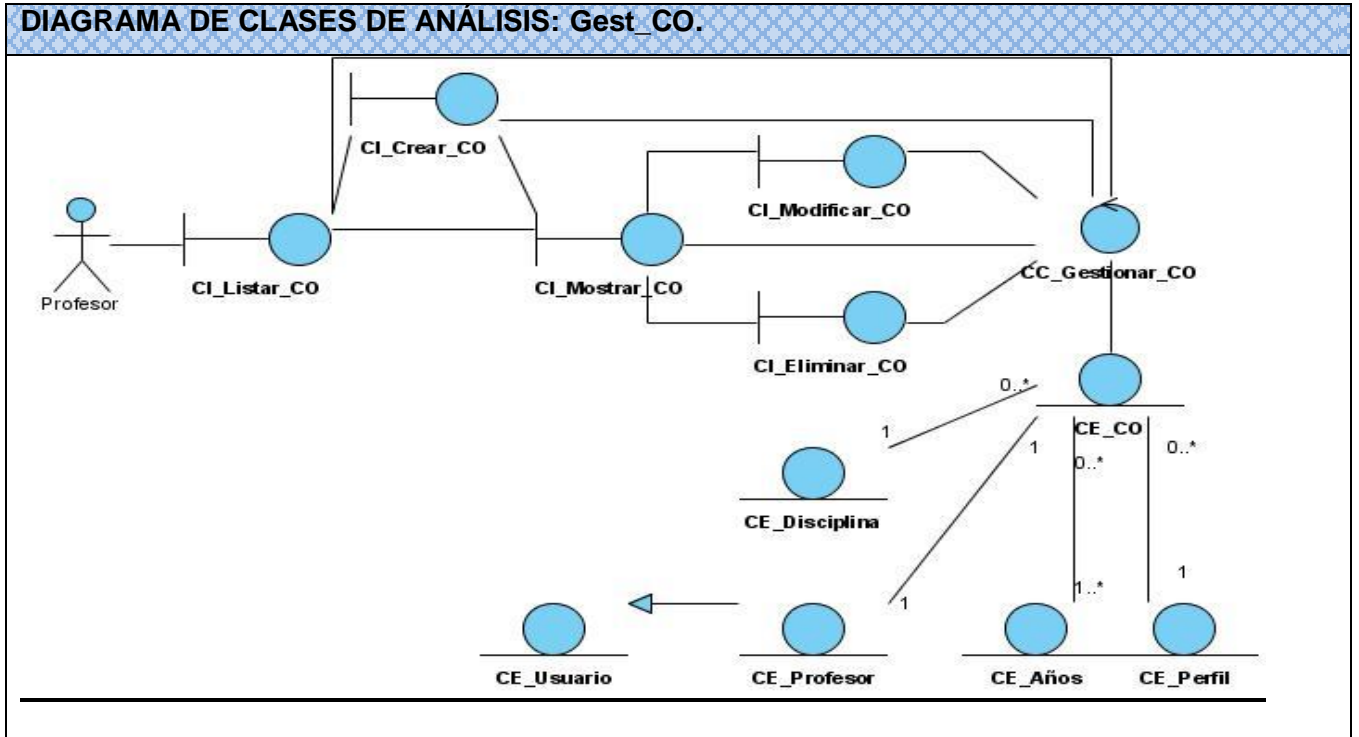


Figura 12: Diagrama de Clases del Análisis Gestionar CO.

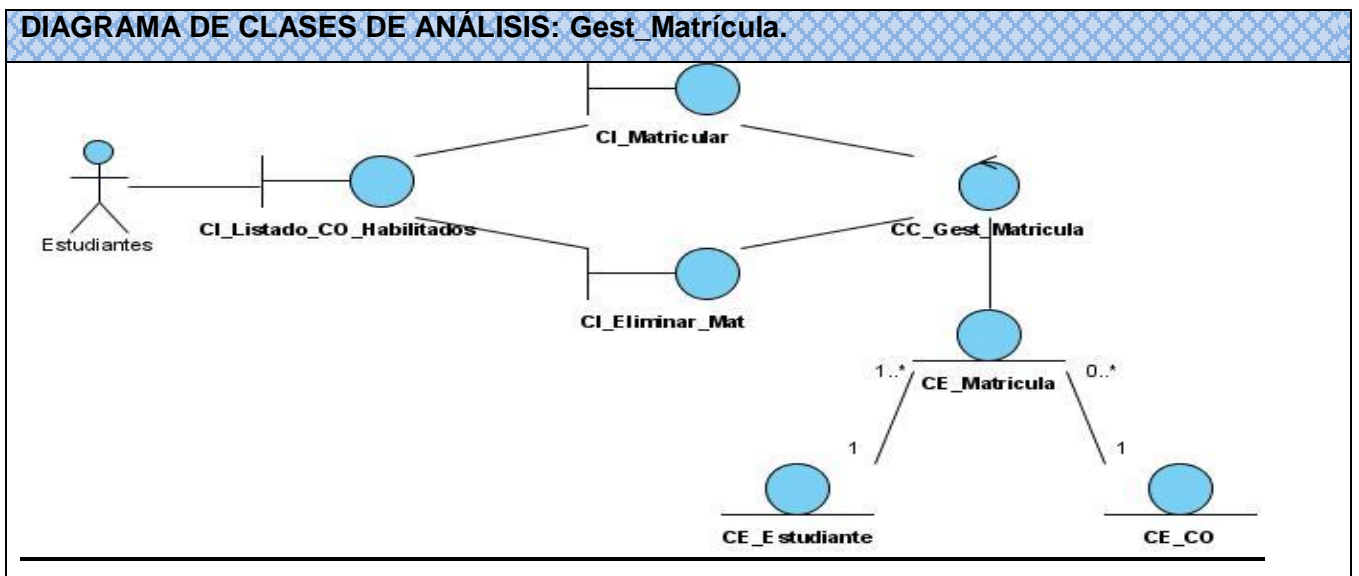


Figura 13: Diagrama de Clases del Análisis Gestionar Matrícula.

El resto de los diagramas de clases del análisis se encuentran en el anexo correspondiente. [Anexo 2]

3.3 Diseño.

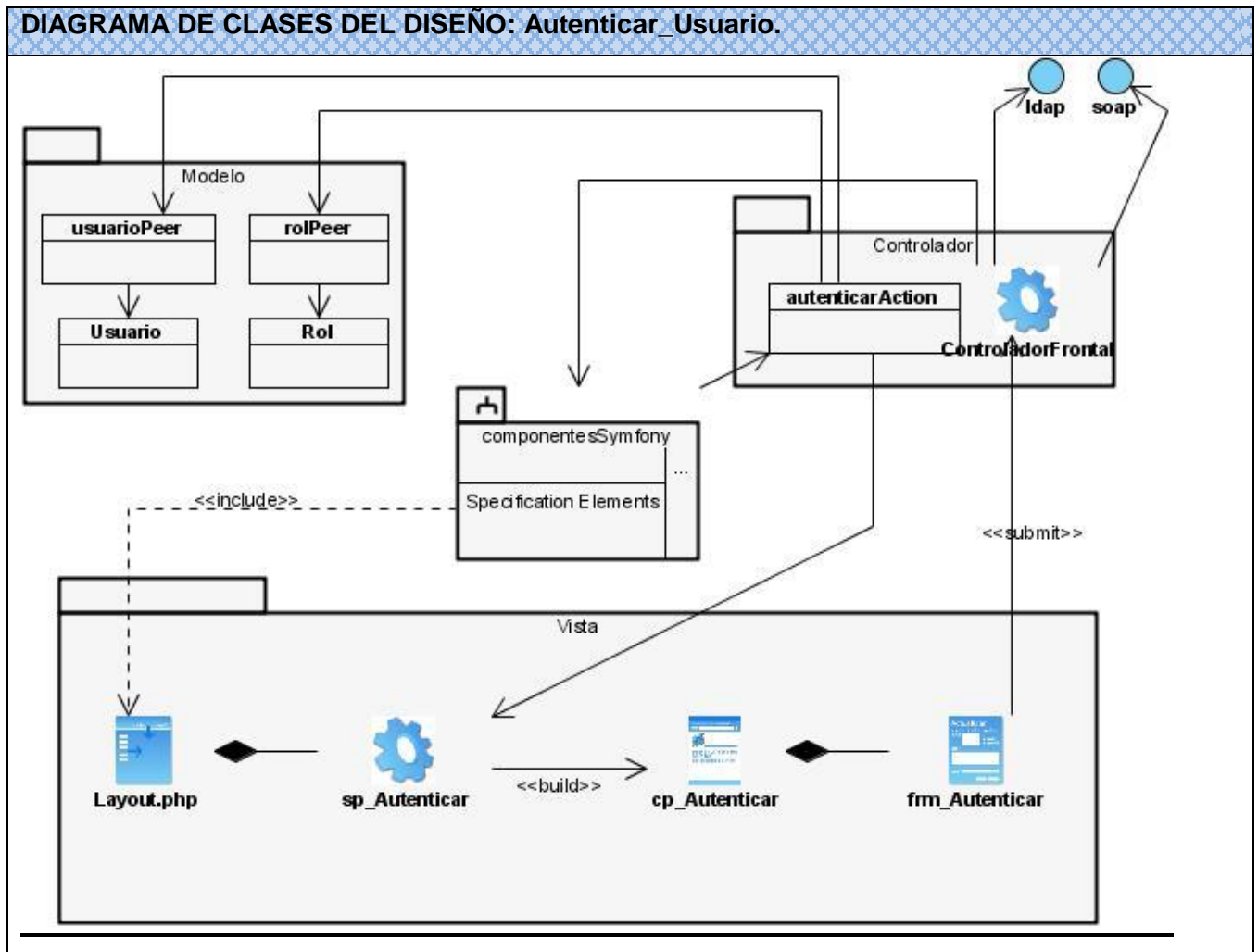


Figura 15: Diagrama de Clases del Diseño Autenticar Usuario.

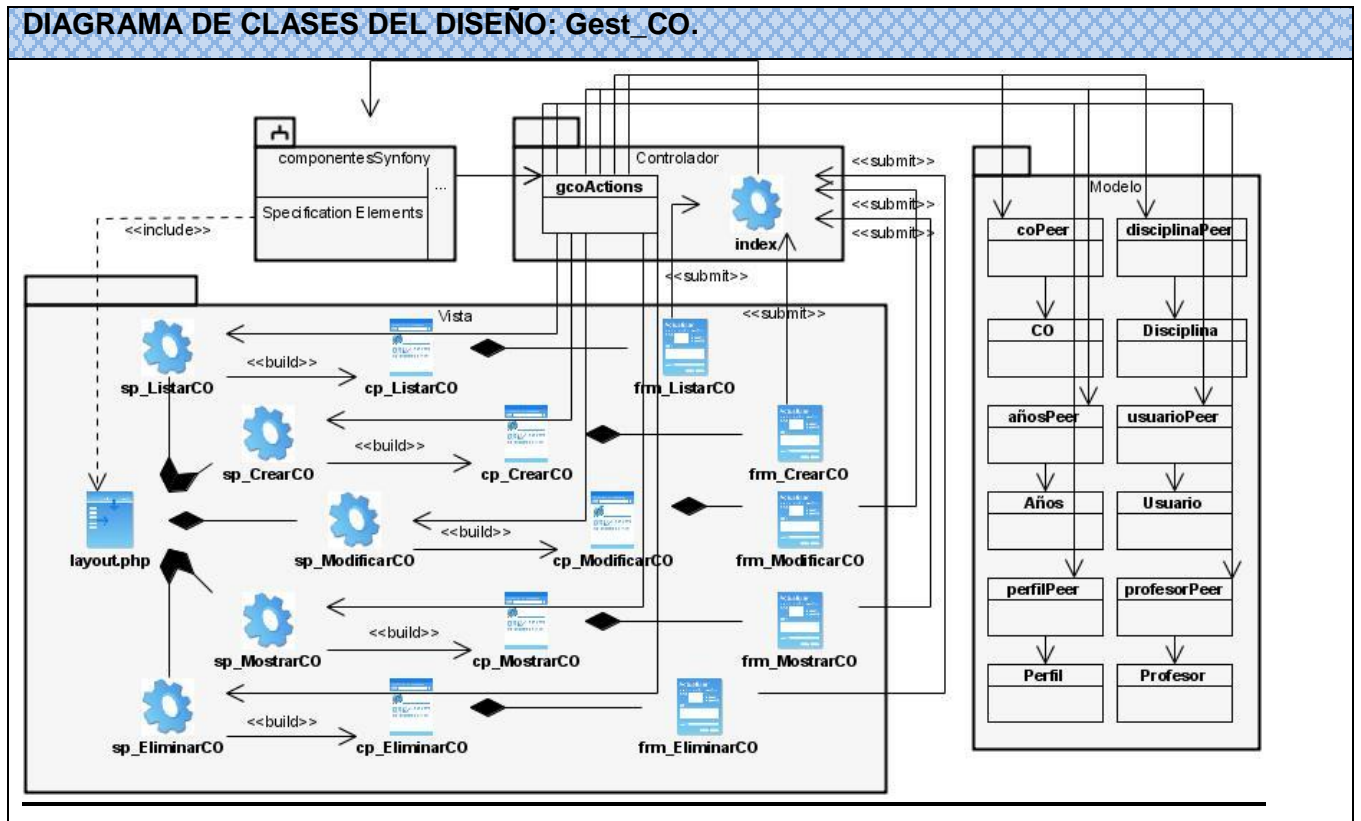


Figura 20: Diagrama de Clases del Diseño Gestionar CO.

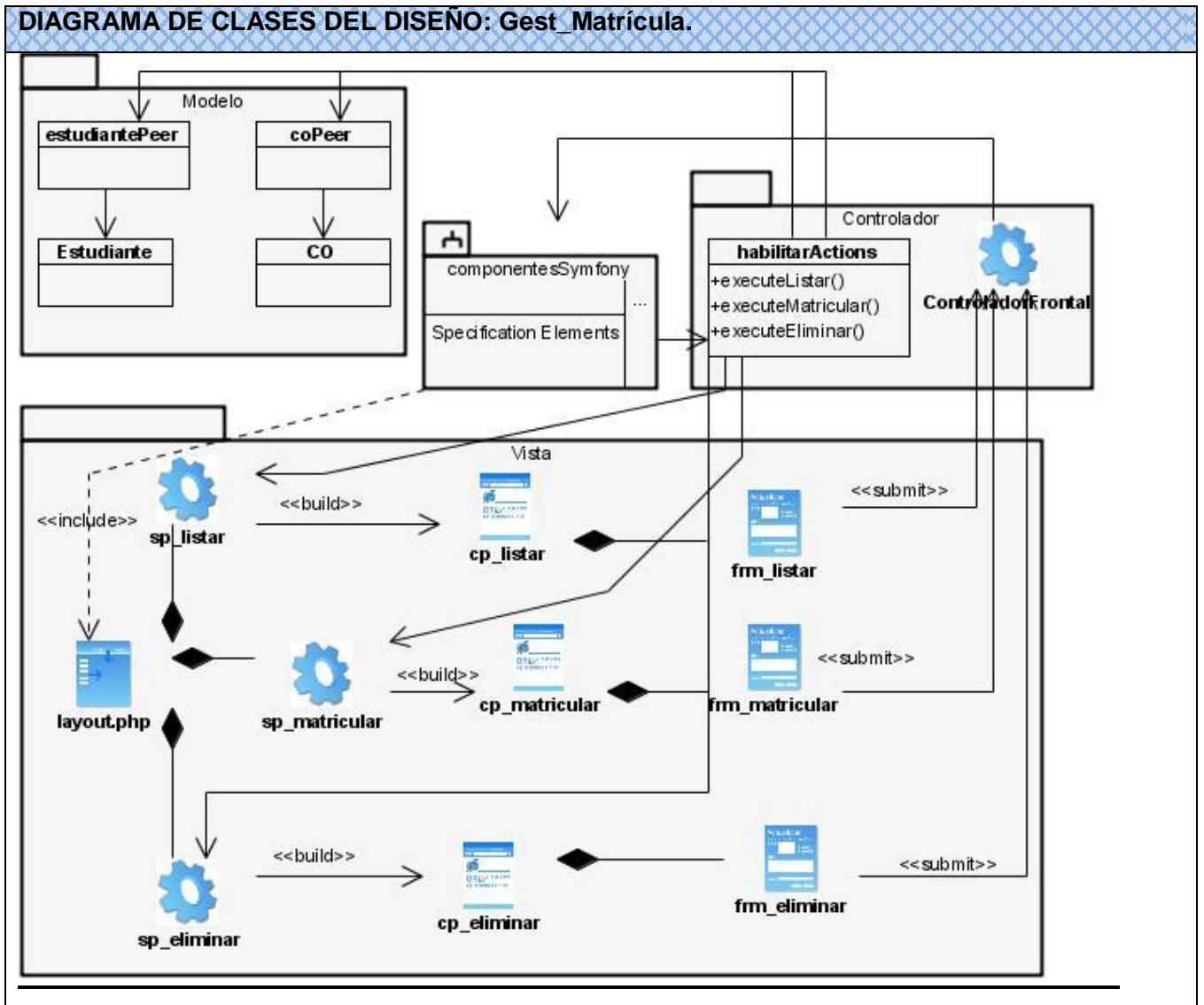


Figura 21: Diagrama de Clases del Diseño Gestionar Matrícula.

El resto de los diagramas de clases del diseño se encuentran en el anexo correspondiente. [Anexo 3]

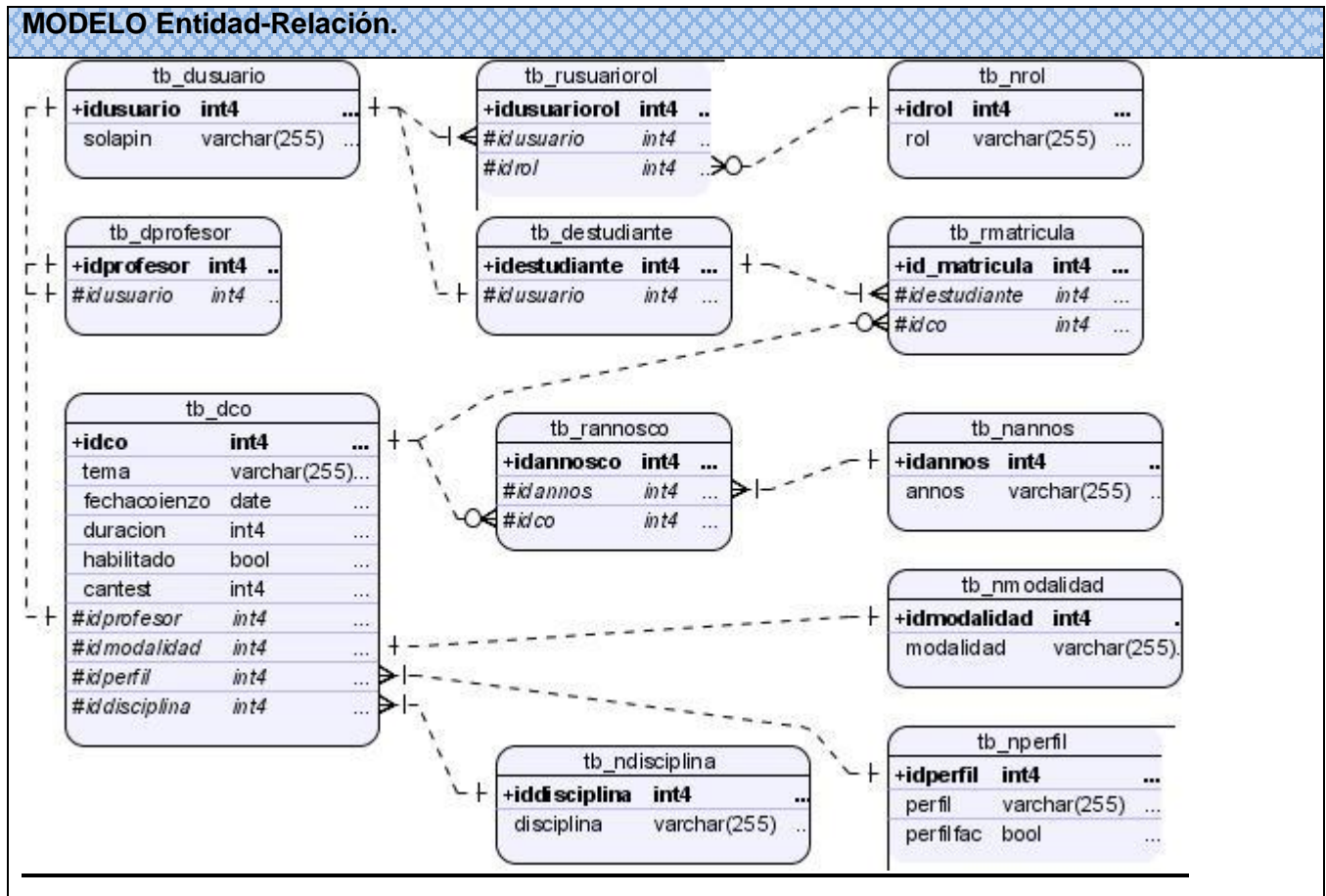


Figura 23: Diagrama Entidad-Relación.

3.4 Descripción de las tablas de la Base de Dato

Tabla 16: Descripción de la tabla Años/CO.

Nombre: Tb_rannosco		
Descripción: Guarda la información de los años de los CO.		
Atributo	Tipo	Descripción
Id_annosco	int	Identificador de la tabla.
Id_annos	int	Hace referencia al identificador de los años.
Id_co	int	Hace referencia al identificador del CO.

Tabla 17: Descripción de la tabla Años.

Nombre: Tb_nannos		
Descripción: Guarda la información de los años de los CO.		
Atributo	Tipo	Descripción
Id_annos	int	Identificador de la tabla.
annos	varchar	Hace referencia a los años.

Tabla 18: Descripción de la tabla Modalidad.

Nombre: Tb_nmodalidad		
Descripción: Guarda la información de las modalidades de los CO.		
Atributo	Tipo	Descripción
Id_modalidad	int	Identificador de la tabla.
modalidad	varchar	Hace referencia a la modalidad del CO.

Tabla 19: Descripción de la tabla Perfil.

Nombre: Tb_nperfil		
Descripción: Guarda la información de las disciplinas de los CO.		
Atributo	Tipo	Descripción
Id_perfil	int	Identificador de la tabla.
perfil	varchar	Hace referencia al perfil del CO.
perfilfac	bool	Hace referencia a si es del perfil de la Facultad.

Tabla 20: Descripción de la tabla Disciplina.

Nombre: Tb_ndisciplina		
Descripción: Guarda la información de las disciplinas de los CO.		
Atributo	Tipo	Descripción
Id_disciplina	int	Identificador de la tabla.
disciplina	varchar	Hace referencia a la disciplina del CO.

Tabla 21: Descripción de la tabla CO.

Nombre: Tb_dco		
Descripción: Guarda la información de los CO.		
Atributo	Tipo	Descripción
Id_co	int	Identificador de la tabla.
tema	varchar	Hace referencia al tema del CO.
fechacomienzo	date	Hace referencia a la fecha de comienzo del CO.
duración	int	Hace referencia a la duración del CO.
habilitado	bool	Hace referencia al estado del CO.
cantest	int	Hace referencia a la cantidad de estudiantes del CO.
Id_profesor	int	Hace referencia al identificador del profesor del CO.
Id_modalidad	int	Hace referencia al identificador de la modalidad del CO.
Id_perfil	int	Hace referencia al identificador del perfil del CO.
Id_disciplina	int	Hace referencia al identificador de la disciplina del CO.

Tabla 22: Descripción de la tabla Matrícula.

Nombre: Tb_matrícula		
Descripción: Guarda la información de las matrículas.		
Atributo	Tipo	Descripción
Id_matrícula	int	Identificador de la tabla.
Id_estudiante	int	Hace referencia al identificador del estudiante.
Id_co	int	Hace referencia al identificador del CO.

Tabla 23: Descripción de la tabla Rol.

Nombre: Tb_nrol		
Descripción: Guarda la información de los roles.		
Atributo	Tipo	Descripción
Id_usuariorol	int	Identificador de la tabla.
rol	varchar	Hace referencia al rol.

Tabla 24: Descripción de la tabla Rol/Usuario.

Nombre: Tb_rusuariorol		
Descripción: Guarda la información de los usuarios y sus roles.		
Atributo	Tipo	Descripción
Id_usuariorol	int	Identificador de la tabla.
Id_rol	int	Hace referencia al identificador del rol.
Id_usuario	int	Hace referencia al identificador del usuario.

Tabla 25: Descripción de la tabla Usuario.

Nombre: Tb_dusuario		
Descripción: Guarda la información de los usuarios.		
Atributo	Tipo	Descripción
Id_usuario	int	Hace referencia al identificador del usuario.
solapín	varchar	Hace referencia al número de solapín del usuario

Tabla 26: Descripción de la tabla Profesor.

Nombre: Tb_dprofesor		
Descripción: Guarda la información del profesor.		
Atributo	Tipo	Descripción
Id_profesor	int	Identificador del profesor
Id_usuario	int	Hace referencia al usuario asociado.

Tabla 27: Descripción de la tabla Estudiante.

Nombre: Tb_destudiante.		
Descripción: Guarda la información de los estudiantes.		
Atributo	Tipo	Descripción
Id_estudiante	int	Identificador del estudiante
Id_usuario	int	Hace referencia al usuario asociado

3.5 Estimación de Esfuerzo.

Una vez determinados los casos de uso que guiarán el desarrollo del software, se puede predecir una estimación del tiempo de duración del proyecto mediante el análisis de Puntos de Casos de Uso.

“La estimación mediante el análisis de Puntos de Casos de Uso se trata de un método de estimación del tiempo de desarrollo de un proyecto mediante la asignación de "pesos" a un cierto número de factores que lo afectan, para finalmente, contabilizar el tiempo total estimado para el proyecto a partir de esos factores.”

A continuación se detallan los pasos a seguir para la realización de este método.

3.5.1 Cálculo de Puntos de Casos de Uso sin ajustar.

El cálculo de Puntos de casos de Uso sin ajustar se calcula mediante la siguiente ecuación:

$$\mathbf{UUCP = UAW + UUCW}$$

Donde:

UUCP: Puntos de Casos de Uso sin ajustar

UAW: Factor de Peso de los Actores sin ajustar

UUCW: Factor de Peso de los Casos de Uso sin ajustar

Factor de Peso de los Actores sin ajustar (UAW)

Este valor se calcula mediante un análisis de la cantidad de actores presentes en el sistema y la complejidad de cada uno de ellos. Los criterios a tener en cuenta se detallan a continuación.

Tabla 28: Factor de Pesos de los Actores sin ajustar.

Tipo de actor	Descripción	Factor de Peso
Simple	Otro sistema que interactúa con el sistema a desarrollar mediante una interfaz de programación.	1
Medio	Otro sistema que interactúa con el sistema a desarrollar mediante un protocolo o una interfaz basada en texto.	2
Complejo	Una persona que interactúa con el sistema mediante una interfaz gráfica.	3

En el sistema propuesto va a interactuar un actor, los usuarios, que constituye un actor complejo ya que es una persona que interactúa con el sistema mediante una interfaz gráfica, por lo tanto el factor de peso sería 3. Luego el factor de peso de los actores sin ajustar se calcula mediante la siguiente ecuación:

$$\text{UAW} = \Sigma \text{Cantidad Actores} * \text{Factor de Peso}$$

$$\text{UAW} = 1 * 3$$

$$\text{UAW} = 3$$

Factor de Peso de los Casos de Uso sin ajustar (UUCW).

El valor del Factor de Peso de los Casos de Uso sin ajustar se calcula mediante un análisis de la cantidad de Casos de Uso presentes en el sistema y la complejidad de cada uno de ellos. La complejidad de los Casos de Uso se establece teniendo en cuenta la cantidad de transacciones efectuadas en el mismo, donde una transacción es una secuencia de actividades atómica, es decir, se efectúa la secuencia de actividades completa, o no se efectúa ninguna de las actividades de la secuencia. Los criterios se muestran en la siguiente tabla:

Tabla 29: Factor de Pesos de los Casos de Uso sin ajustar.

Tipo de Caso de Uso	Descripción	Factor de peso
Simple	El caso de Uso contiene de 1 a 3 transacciones.	5
Medio	El caso de Uso contiene de 4 a 7 transacciones.	10
Complejo	El caso de Uso contiene más de 8 transacciones.	15

Para la realización de la aplicación se tienen en cuenta 8 casos de uso, los cuales están de 1 a 3 transacciones, por lo que todos los casos de uso tienen un Tipo "Simple" con un "Factor de peso" igual a 5, entonces el factor de peso de los casos de uso sin ajustar sería:

$$\text{UUCW} = \Sigma \text{Cantidad CU} * \text{Factor de Peso}$$

$$\text{UUCW} = 8 * 5$$

$$\text{UUCW} = 40$$

Por tanto los Puntos de Casos de Uso sin ajustar sería:

$$\text{UUCP} = \text{UAW} + \text{UUCW}$$

UUCP = 3 + 40

UUCP = 43

3.5.2 Cálculo de los Puntos de Casos de Uso ajustados.

Ya obtenidos los Puntos de casos de uso sin ajustar, se debe ajustar este valor mediante la siguiente ecuación:

UCP = UUCP x TCF x EF

Donde:

UCP: Puntos de Casos de Uso ajustados

UUCP: Puntos de Casos de Uso sin ajustar

TCF: Factor de complejidad técnica

EF: Factor de ambiente

El factor de complejidad técnica (TCF) se calcula mediante la cuantificación de un conjunto de factores que determinan la complejidad técnica del sistema. Cada factor se cuantifica en un valor desde 0 a 5, donde 0 sería un aporte irrelevante y 5 sería un aporte muy relevante.

Tabla 30: Factor de complejidad técnica.

Factor	Descripción	Peso	Valor asignado	Peso * Valor asignado
T1	Sistema distribuido	2	0	0
T2	Objetivos de performance o tiempo de respuesta.	1	4	4
T3	Eficiencia del usuario final.	1	4	4
T4	Procesamiento interno complejo.	1	3	3
T5	El código debe ser reutilizable.	1	3	3
T6	Facilidad de instalación.	0.5	3	1.5
T7	Facilidad de uso.	0.5	2	1
T8	Portabilidad.	2	0	0
T9	Facilidad de cambio.	1	1	1
T10	Concurrencia.	1	0	0

Capítulo 3: Análisis y Diseño.

T11	Incluye objetivos especiales de seguridad.	1	3	3
T12	Provee acceso directo a terceras partes.	1	0	0
T13	Se requieren facilidades especiales de entrenamiento a usuarios.	1	1	1

$$\text{TCF} = 0.6 + 0.01 * \Sigma (\text{Peso} \times \text{Valor asignado})$$

$$\text{TCF} = 0.6 + 0.01 * 21.5$$

$$\text{TCF} = 0.815 \sim 0.82.$$

El factor de ambiente (EF) está relacionado con las habilidades y entrenamiento del grupo de desarrollo que realiza el sistema. Cada factor se cuantifica con un valor desde 0 a 5.

Tabla 31: Factor de ambiente.

Factor	Descripción	Peso	Valor Asignado	Peso * Valor asignado
E1	Familiaridad con el modelo del proyecto utilizado.	1.5	2	3
E2	Experiencia en la aplicación.	0.5	2	1
E3	Experiencia en orientación a objetos.	1	3	3
E4	Capacidad del analista líder.	0.5	1	0.5
E5	Motivación.	1	1	1
E6	Estabilidad de los requerimientos.	2	4	8
E7	Personal part-time.	-1	1	-1
E8	Dificultad del lenguaje de programación.	-1	2	-2

$$\text{EF} = 1.4 - 0.03 * \Sigma (\text{Peso} * \text{Valor asignado})$$

$$\text{EF} = 1.4 - 0.03 * 13.5$$

$$\text{EF} = 0.995$$

Calculando los puntos de Caso de Uso Ajustados quedaría:

$$\text{UCP} = \text{UUCP} * \text{TCF} * \text{EF}$$

$$\text{UCP} = 103 * 0.82 * 0.995$$

$$\text{UCP} = 84.0377 \sim 84.04$$

3.5.3 Estimación de esfuerzo a través de los puntos de casos de uso.

Que está dado por la siguiente ecuación:

$$E = \text{UCP} * \text{CF}$$

Donde:

E: Esfuerzo estimado en horas hombres.

UCP: Punto de casos de usos ajustados.

CF: Factor de conversión.

Para calcular el **Factor de conversión** se tiene en cuenta:

Se contabilizan cuántos factores de los que afectan al Factor de ambiente están por debajo del valor medio (3), para los factores E1 a E6.

Se contabilizan cuántos factores de los que afectan al Factor de ambiente están por encima del valor medio (3), para los factores E7 y E8.

Como el total contabilizado es 2 se utiliza el **factor de conversión 20 horas-hombre/Punto de Casos de Uso.**

$$E = \text{UCP} * \text{CF}$$

$$E = 84.04 * 20$$

$$E = 1680.8 \text{ Horas Hombre}$$

3.5.4 Calcular el esfuerzo de todo el proyecto.

Tabla 32: Esfuerzo del Proyecto

Actividad	Porcentaje	Esfuerzo
Análisis	25.00 %	2101
Diseño	20.00%	1680.8
Programación	20.00 %	1680.8
Pruebas	25.00 %	2101

Sobrecarga(otras actividades)	10.00%	840.4
Total	100.00 %	8404

El proyecto requiere de 2101 horas-hombre para su desarrollo. Trabajando 12 horas diarias se obtiene aproximadamente 175.80 días para el desarrollo del proyecto, es decir, para un equipo de 1 personas trabajando 26 días al mes el proyecto tiene una duración de 6 meses y medio (6.73 meses-hombre).

3.6 Conclusiones.

En este capítulo se realizó el análisis y diseño donde se relaciona los requisitos funcionales y los no funcionales, llevándolo a una expresión de la programación entre las diferentes herramientas mencionadas en capítulos anteriores y algunos aspecto importante para la necesidad del usuario, a su vez la preparación para la etapa de implementación del sistema. Además se realizó el análisis del esfuerzo del proyecto, donde pudimos ver la duración y costo del mismo.

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 de 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.

4.2.1. Diagrama de Despliegue.

El Diagrama de Despliegue muestra las relaciones que se establecen entre los componentes de software y hardware, representado mediante nodos estrechamente conectados. Como se puede observar se utiliza un Servidor de Base de Datos que contiene la información que se trata. Este servidor está conectado al Servidor de Aplicaciones Web, y este a su vez está conectado a la PC Cliente que representa a todas las PC donde se va a utilizar el sistema, las que deben cumplir con los requisitos de hardware necesarios para que el sistema funcione correctamente.

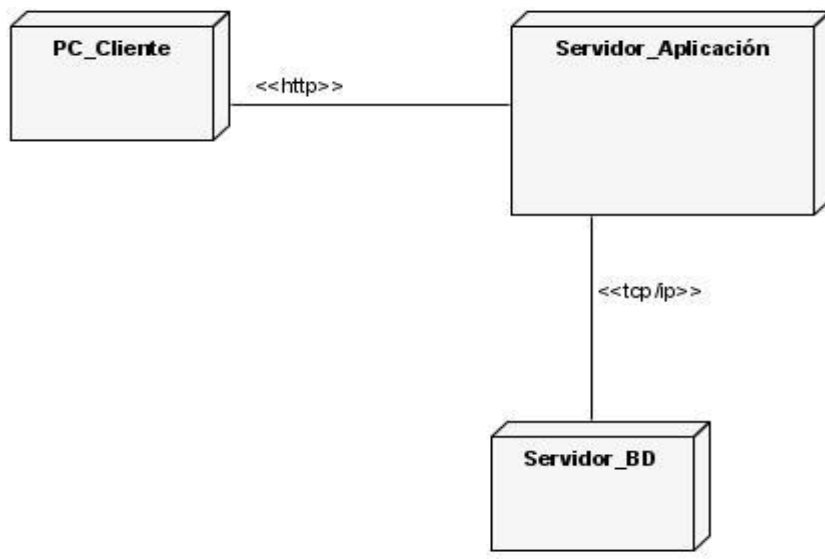


Figura 24: Diagrama de Despliegue.

4.2.2. Diagramas de Componentes.

Los Diagramas de Componentes muestran la relación entre los elementos de la implementación. Se representan mediante grafos relacionados por relaciones de compilación y ejecución.

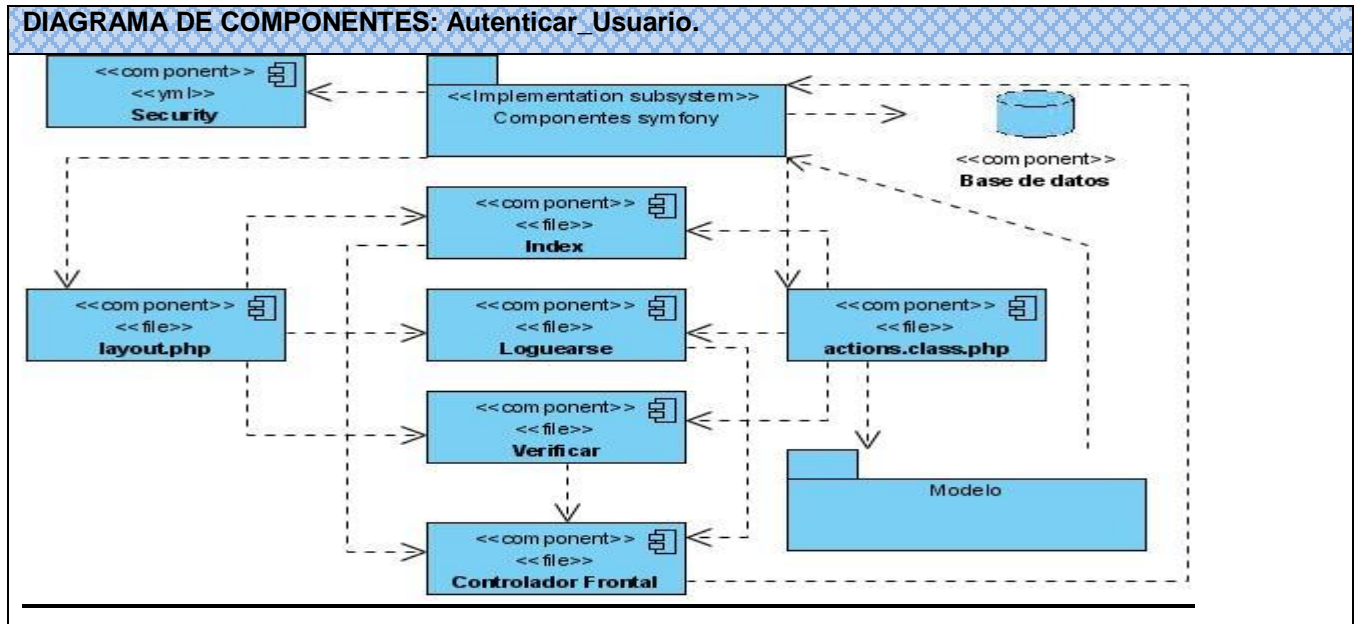


Figura 25: Diagrama de Componentes Autenticar Usuario.

El resto de los diagramas de componentes se encuentran en el anexo correspondiente. [Anexo 4]

4.3. Modelo de 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.

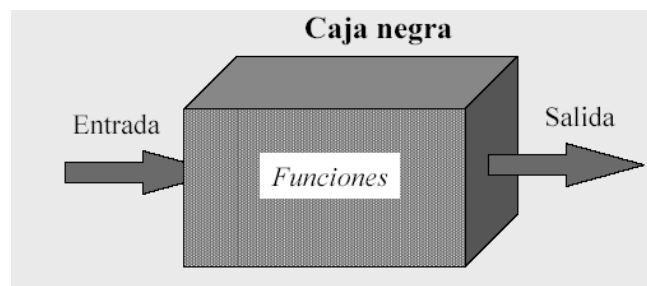


Figura 40: Representación de la Prueba de Caja Negra.

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.

Capítulo 4: Implementación y Prueba.

4.3.1. Casos de Prueba

Tabla 33: Caso de Prueba: Crear Curso Optativo.

Condición de Entrada	Casos Válidos	Casos no Válidos
El usuario debe introducir el tema, la matrícula y la duración del CO.	Que el usuario introduzca el tema, la matrícula y la duración del CO.	Que el usuario no introduzca alguno de los parámetros.
El usuario debe seleccionar el perfil, la disciplina y la modalidad del CO.	Que el usuario seleccione el perfil, la disciplina y la modalidad del CO.	Que el usuario no seleccione alguno de los datos.
El usuario debe seleccionar la fecha de comienzo del CO.	La fecha seleccionada sea superior a la fecha actual.	La fecha seleccionada no sea superior a la fecha actual.

Caso de uso: Gestionar CO.
Caso de Prueba: Crear Curso Optativo.
Entrada: Tema: Tema Curso Optativo. Fecha: 4/6/2009. Duración: 20 h/c. Perfil: Perfil1. Disciplina: Disciplina1. Modalidad: Presencial. Matrícula: 25.
Resultado: El sistema adiciona el curso a la base de datos y actualiza el listado de CO no habilitados.
Condiciones: Los datos para crear el CO deben estar en el rango de valores válidos.

Caso de uso: Gestionar CO.

Capítulo 4: Implementación y Prueba.

Caso de Prueba: Crear Curso Optativo.
Entrada: Tema: Fecha: 14/6/2009. Duración: 20 h/c. Perfil: Disciplina: Disciplina1. Modalidad: Presencial. Matrícula: 25.
Resultado: El sistema muestra un mensaje de Error.
Condiciones: Todos los campos deben estar llenos y la fecha de inicio no puede ser mayor que la fecha actual.

Tabla 34: Caso de Prueba: Matricularse en curso optativo.

Condición de Entrada	Casos Válidos	Casos no Válidos
El usuario debe seleccionar el CO.	El usuario seleccione el CO.	El usuario no seleccione el CO.
El usuario debe seleccionar la opción matricularse.	El usuario no debe haberse matriculado anteriormente.	El usuario se halla matriculado con anterioridad.

Caso de uso: Matricular.
Caso de Prueba: Matricularse en curso optativo.
Entrada: Curso: Curso Optativo 1. Estudiante: 50074.
Resultado: El sistema matricula al estudiante en el curso seleccionado.
Condiciones: El estudiante seleccione el curso y no se halla matriculado con anterioridad.

Capítulo 4: Implementación y Prueba.

Caso de uso: Matricular.
Caso de Prueba: Matricularse en curso optativo.
Entrada: Curso: Curso Optativo 1. Estudiante: 50074.
Resultado: El sistema muestra un mensaje de Error.
Condiciones: El usuario no puede estar previamente matriculado en el curso seleccionado.

Tabla 35: Caso de Prueba: Matrícula Forzada.

Condición de Entrada	Casos Válidos	Casos no Válidos
El usuario debe seleccionar el CO.	El usuario seleccione el CO.	El usuario no seleccione el CO.
El usuario debe introducir el número de solapín.	El número de solapín sea correcto y el propietario no se halla matriculado en dicho curso con anterioridad.	El número de solapín sea incorrecto o el propietario se halla matriculado con anterioridad.

Caso de uso: Matrícula Forzada.
Caso de Prueba: Realizar Matrícula Forzada.
Entrada: Solapín: 50074.
Resultado: El sistema matricula al usuario del solapín en el CO seleccionado.
Condiciones: El número de solapín tiene que ser correcto (pertenecer al dominio) y no estar registrada la matrícula.

Caso de uso: Matrícula Forzada.
Caso de Prueba: Realizar Matrícula Forzada.
Entrada:

Capítulo 4: Implementación y Prueba.

Solapín: 50074.
Resultado: El sistema muestra un mensaje de Error.
Condiciones: El usuario del solapín no puede estar previamente matriculado.

Caso de uso: Matrícula Forzada.
Caso de Prueba: Realizar Matrícula Forzada.
Entrada: Solapín: 50A07411.
Resultado: El sistema muestra un mensaje de Error.
Condiciones: El número de solapín debe ser correcto (debe pertenecer al dominio).

4.4. Conclusiones.

En este capítulo se tratan los flujos de trabajo de Implementación y Prueba. En la Implementación tenemos los Diagramas de Componentes y Despliegue. En los procesos de pruebas, se encuentran el diseño de los Casos de Pruebas que demuestran hasta cierto punto como se debe probar el sistema. Con el diseño de los elementos principales de ambos flujos se cumplen los objetivos trazados para el desarrollo de este capítulo.

Conclusiones Generales.

Con la realización del presente trabajo:

- Se desarrolló una aplicación capaz de gestionar los cursos optativos y el proceso de matrícula de los estudiantes en los mismos, así como, las matrículas forzadas que por interés o necesidad de la facultad se orienten.
- Se describieron los procesos, lo cual brindó una mayor claridad a la hora de implementar el sistema.
- Se seleccionaron las herramientas más adecuadas para el desarrollo del Sistema de Gestión de los Cursos Optativos de la Facultad 1 de la Universidad de las Ciencias Informáticas, luego de haber realizado un estudio de las tendencias actuales en el desarrollo de aplicaciones web.
- Se realizó el levantamiento de requisitos funcionales y no funcionales, a partir de los cuales se definieron y describieron cada una de las funcionalidades que posee el sistema.
- Con la descripción de los Casos de Prueba se probó la fiabilidad y el rendimiento de la aplicación.
- Se pusieron en práctica los conocimientos adquiridos durante la carrera en las disciplinas de Ingeniería de Software y Programación.

Por tanto, podemos concluir que se logró satisfactoriamente el cumplimiento del objetivo propuesto para el presente trabajo de diploma.

Recomendaciones.

- Se implementen nuevas funcionalidades que aporten comodidad, facilidad y resuelvan nuevas peticiones de los implicados según necesidades o interés de los mismos como son: un mecanismo para recoger la asistencia y las notas de los estudiantes en los cursos optativos, así como archivar los mismos para tener un control de los curso que ha pasado cada estudiante y la nota obtenida en los mismos.
- Implementar una traza de seguridad en el proceso de gestión de los cursos optativos.
- La Facultad conserve el documento como consulta y guía para aquellos que darán mantenimiento al sistema.

Referencias Bibliográficas.

- [1] ACHOUR, M.; F. BETZ, *et al.* Manual de PHP, 2009. [Disponible en: <http://es.php.net/manual/es/index.php>]
- [2] François Zaninotto y Fabien Potencier, t. p. Symfony, la guía definitiva. s.l. : Apress, 2008.
- [3] IVAR JACOBSON, G. B., JAMES RUMBAUGH. *El Proceso Unificado de Desarrollo de Software*. 2da Edición. Madrid, España, Adison Wesley, 2000. 464 p
- [4] Booch, G.: Rumbaugh, J. y Jacobson. "El Lenguaje Unificado de Modelado". s.l. Addison-Wesley.
- [5] *Trabajo de Auditoria "Herramientas Case"*. [En Línea] 17 [Citado el 2 de febrero de 2009] Disponible en: http://html.rincondelvago.com/herramientas-case_2.html
- [6] Free Download Manager. [En línea] [Citado el: 20 de enero de 2009.] Disponible en [http://www.freedownloadmanager.org/es/downloads/Paradigma_Visual_para_UML_\(M%C3%8D\)_14720_p/](http://www.freedownloadmanager.org/es/downloads/Paradigma_Visual_para_UML_(M%C3%8D)_14720_p/).
- [7] Pecos, D. (n.d.). *PostGreSQL*. [En Línea] [Citado el 17 de febrero de 2009] Disponible en: http://www.netpecos.org/docs/mysql_postgres/index.html
- [8] E.V.A. UCI, I. D. S. Conferencia #2. Modelación de negocio, 2009. ISW 1 [Disponible en: <http://teleformacion.uci.cu/mod/resource/view.php?id=11569>]
- [9] Estructura de la UJC en la UCI [Disponible en: <http://ujc.uci.cu/tematicas/ujc-en-la-uci/estructura/>]
- [10] Entorno Virtual de Aprendizaje Conf. 4 FT Requerimiento [En Línea] [Citado el 18 de marzo de 2009] [Disponible en: <http://teleformacion.uci.cu/mod/resource/view.php?id=12075>]
- [11] Entorno Virtual de Aprendizaje, Conferencia 3 FT de Análisis, 2008 – 2009. [Disponible en: <http://teleformacion.uci.cu/mod/resource/view.php?id=13151>]
- [12] Entorno Virtual de Aprendizaje, Conferencia 2 Arquitectura y Patrones de diseño, 2008 – 2009. [Disponible en: <http://teleformacion.uci.cu/mod/resource/view.php?id=14077>]
- [13] Mendoza Navarro, Javier. Diseño del Sistema, 2009. [Disponible en: <http://teleformacion.uci.cu/mod/resource/view.php?id=14074>]

Bibliografía.

1. Yanoski Calderín Delgado, GESTACAD. Sistema para la Gestión Académica. [Online] [Citado el 10 de febrero del 2009.] [Disponible en: <http://www.posgrados.frc.utn.edu.ar/congreso/trabajos/7.doc>
2. SCADA. Sistema de Supervisión de Control y Adquisición de Datos. [Online] [Citado el 12 de febrero del 2009.] [Disponible en: http://bibliodoc.uci.cu/TD/TD_1611_08.pdf
3. Manual de PHP. ACHOUR, M.; F. BETZ, [Online] [Citado el 10 de febrero del 2009.] [Disponible en: <http://es.php.net/manual/es/index.php>
4. Elementos de UML. [Online] [Citado el 13 febrero del 2009.] [Disponible en: <http://docs.kde.org/kde3/es/kdesdk/umbrello/uml-elements.html>.
5. Visual Paradigm – UML. [Online] [Citado el 13 de febrero del 2009.] [Disponible en: [http://www.freedownloadmanager.org/es/downloads/Paradigma_Visual_para_UML_\(M%C3%8D\)_14720_p/](http://www.freedownloadmanager.org/es/downloads/Paradigma_Visual_para_UML_(M%C3%8D)_14720_p/)
6. Pecos, D. (n.d.). *PostGreSQL*. [En Línea] [Citado el 17 de febrero de 2009] Disponible en: http://www.netpecos.org/docs/mysql_postgres/index.html
7. Jesús García Molina. De los Procesos del Negocio a los Casos de Uso. [Online] [Citado el 24 de febrero del 2009.] [Disponible en: <http://dis.um.es/~jmolina/jis2000modeladonegocio.pdf>.
8. Requerimientos Funcionales y No Funcionales. [Online] [Citado el 17 de marzo del 2009.] [Disponible en: <http://www.mitecnologico.com/Main/RequerimientosFuncionalesYNoFuncionales>.
9. Elementos de UML. [Online] [Citado el 17 de marzo del 2009.] [Disponible en: <http://docs.kde.org/kde3/es/kdesdk/umbrello/uml-elements.html>.
10. Diagramas de casos de uso. [Online] [Citado el 10 de marzo del 2009.] [Disponible en: <http://tvdi.det.uvigo.es/~avilas/UML/node25.html>.

Glosario de Términos.

CUN: Caso de uso del negocio.

CUS: Caso de uso del sistema.

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.

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

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.

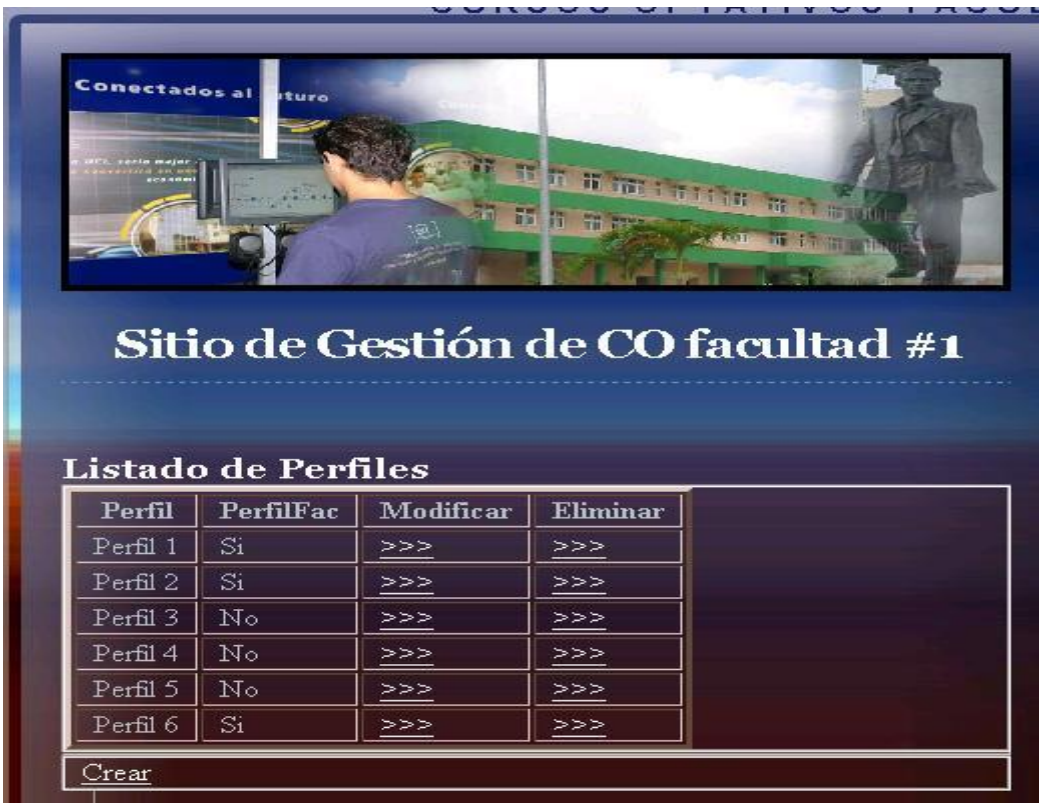
UML: *Unified Modeling Language*. Es una notación estándar para modelar objetos del mundo real como primer paso en el desarrollo de programas orientados a objetos. Es un lenguaje para visualizar, especificar, construir y documentar los artefactos de un sistema de software.

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

Anexos.

Anexo1. Descripción de los Casos de Uso del Sistema.

Tabla 10: Descripción textual del CUS Gestionar Perfil.

CU-3	Gestionar Perfil
Actores	Administrador
Propósito: Permitir al administrador gestionar los Perfiles de los CO del sistema.	
Resumen: El caso de uso se inicia cuando el administrador del sistema desea adicionar, modificar o eliminar algún perfil. El caso de uso termina cuando se realiza algunas de estas acciones.	
Interfaz	
	
Flujo Normal de los Eventos	
Sección "Adicionar Perfil"	
Acción del actor	Respuesta del sistema

1. El administrador selecciona la opción gestionar perfil.	2. Se muestra el listado de los perfiles existentes.
3. Selecciona la opción adicionar.	4. Muestra los campos a llenar.
5. Introduce los datos en los campos.	7. Se validan los datos entrados.
6. Presiona el botón aceptar.	

Flujos alternativos

Acción del actor	Respuesta del sistema
	7.1. Si alguno de los datos entrados es incorrecto, se muestra un mensaje de error y se permite entrar los datos nuevamente. 7.2. Si el rol insertado ya existe se muestra un mensaje y se muestra el listado de los Roles existentes.

Interfaz I



Flujo Normal de los Eventos

Sección "Modificar Perfil"

Acción del actor	Respuesta del sistema
8. El administrador selecciona la opción gestionar perfil.	9. Se muestra el listado de los perfiles existentes.
10. Selecciona la opción modificar.	11. Muestra los datos del perfil a modificar.
12. Modifica los datos en los campos.	14. Se validan los datos modificados.
13. Presiona el botón aceptar.	

Flujos alternativos

Acción del actor	Respuesta del sistema

	14.1. Si alguno de los datos entrados es incorrecto, se muestra un mensaje de error y se permite entrar los datos nuevamente.
--	---

Interfaz II



Flujo Normal de Eventos

Sección "Eliminar Perfil"

Acción del actor	Respuesta del sistema
15. El administrador selecciona la opción gestionar perfil.	16. Se muestra el listado de los perfiles existentes.
17. Selecciona la opción eliminar.	18. Muestra mensaje de confirmación.
19. Presiona el botón aceptar.	20. Se elimina el rol seleccionado.

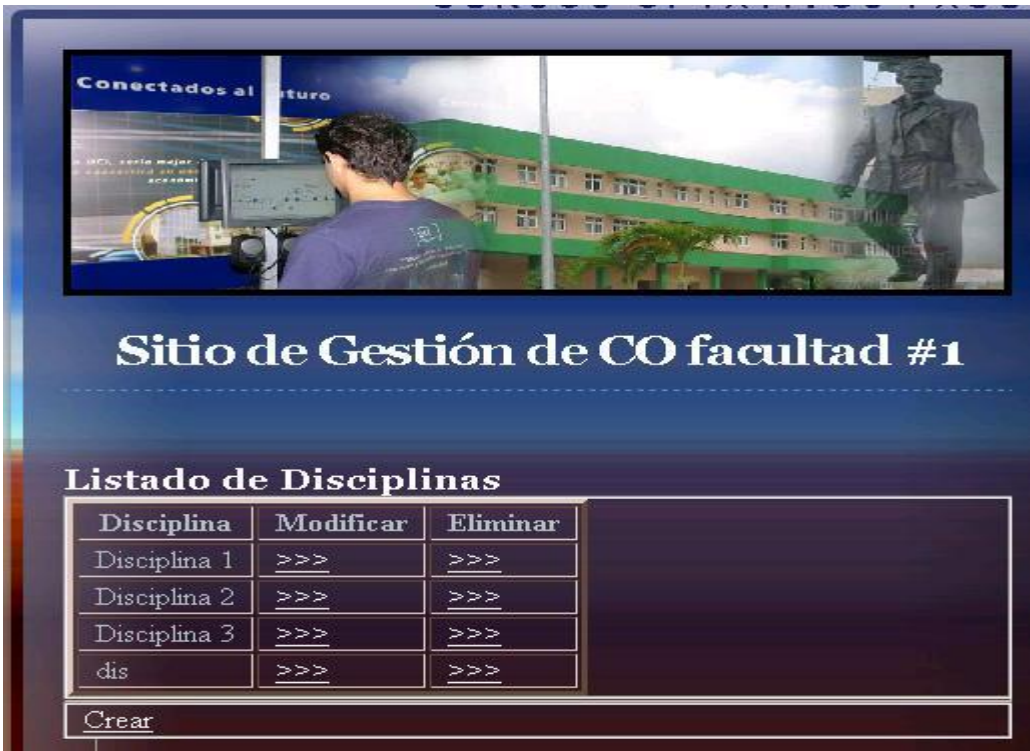
Flujos alternativos

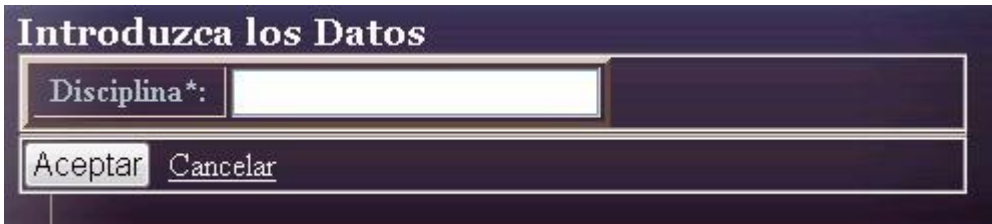
Acción del actor	Respuesta del sistema
19.1. Presiona el botón cancelar.	20. Se vuelve a mostrar el listado de los Perfiles existentes.

Interfaz III



Tabla 11: Descripción textual del CUS Gestionar Disciplina.

CU-4	Gestionar Disciplina	
Actores	Administrador	
Propósito: Permitir al administrador gestionar las disciplinas de los CO del sistema.		
Resumen: El caso de uso se inicia cuando el administrador del sistema desea adicionar, modificar o eliminar disciplinas. El caso de uso termina cuando se realiza algunas de estas acciones.		
Interfaz		
		
Flujo Normal de Eventos		
Sección "Adicionar Disciplina"		
Acción del actor	Respuesta del sistema	
1. El administrador selecciona la opción gestionar disciplina.	2. Se muestra el listado de las disciplinas existentes.	
3. Selecciona la opción adicionar.	4. Muestra los campos a llenar.	
5. Introduce los datos en los campos.	7. Se validan los datos entrados.	

6. Presiona el botón aceptar.	
Flujos alternativos	
Acción del actor	Respuesta del sistema
	<p>7.1. Si alguno de los datos entrados es incorrecto, se muestra un mensaje de error y se permite entrar los datos nuevamente.</p> <p>7.2. Si la disciplina insertada ya existe se muestra un mensaje y se muestra el listado de las disciplinas existentes.</p>
Interfaz I	
	
Flujo Normal de Eventos	
Sección "Modificar Disciplina"	
Acción del actor	Respuesta del sistema
8. El administrador selecciona la opción gestionar disciplina.	9. Se muestra el listado de las disciplinas existentes.
10. Selecciona la opción modificar.	11. Muestra los datos de la disciplina a modificar.
12. Modifica los datos en los campos. 13. Presiona el botón aceptar.	14. Se validan los datos modificados.
Flujos alternativos	
Acción del actor	Respuesta del sistema
	14.1. Si alguno de los datos entrados es incorrecto, se muestra un mensaje de error y se permite entrar los datos nuevamente.
Interfaz II	


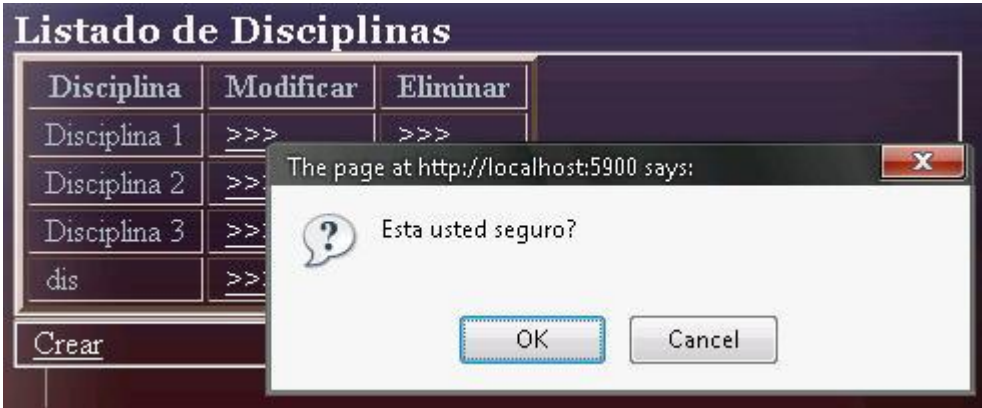
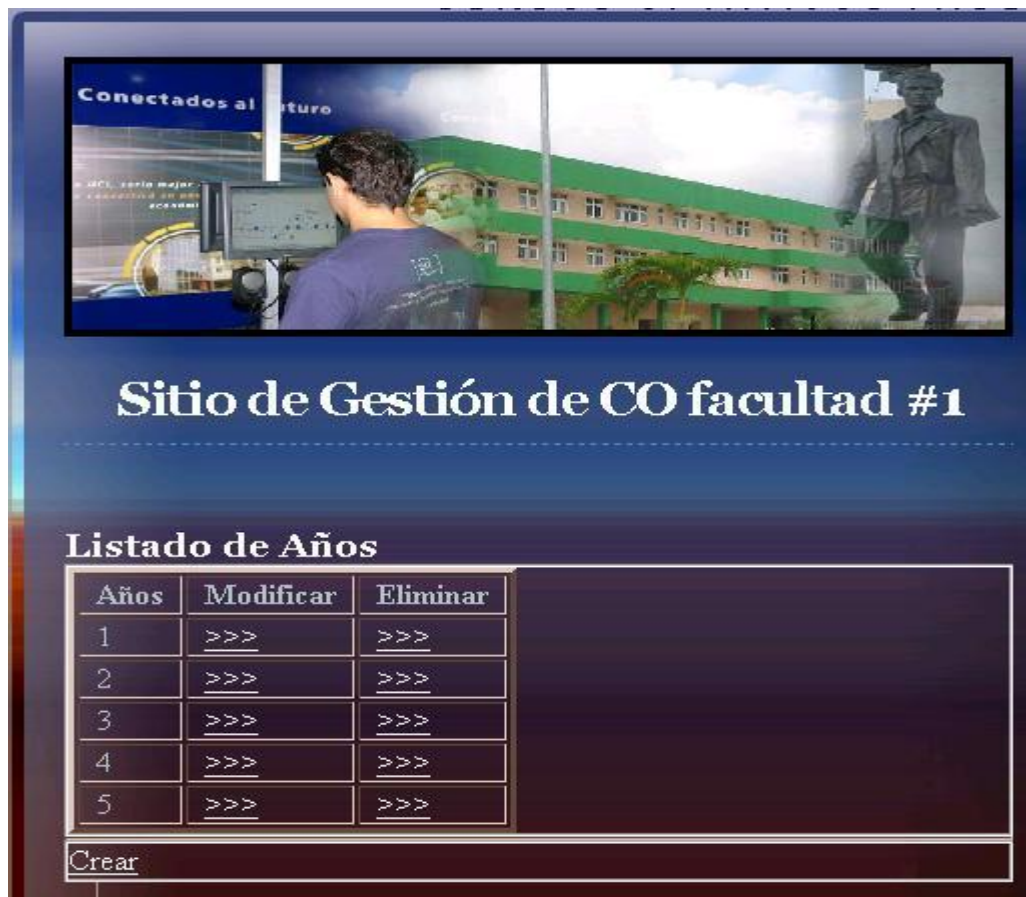
	
Flujo Normal de Eventos	
Sección "Eliminar Disciplina"	
Acción del actor	Respuesta del sistema
15. El administrador selecciona la opción gestionar disciplina.	16. Se muestra el listado de las disciplinas existentes.
17. Selecciona la opción eliminar.	18. Muestra mensaje de confirmación.
19. Presiona el botón aceptar.	20. Se elimina la disciplina seleccionada.
Flujos alternativos	
Acción del actor	Respuesta del sistema
19.1. Presiona el botón cancelar.	20. Se vuelve a mostrar el listado de las Disciplinas existentes.
Interfaz III	
	

Tabla 12: Descripción textual del CUS Gestionar Año.

CU-5	Gestionar Año
Actores	Administrador
Propósito: Permitir al administrador gestionar los años de los CO del sistema.	

Resumen: El caso de uso se inicia cuando el administrador del sistema desea adicionar, modificar o eliminar años. El caso de uso termina cuando se realiza algunas de estas acciones.

Interfaz

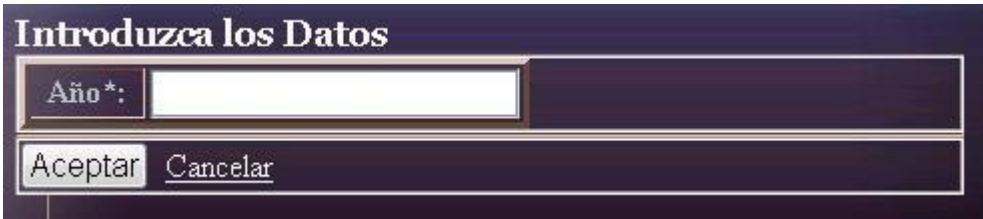


Flujo Normal de Eventos

Sección "Adicionar Año"

Acción del actor	Respuesta del sistema
1. El administrador selecciona la opción gestionar año.	2. Se muestra el listado de los años existentes.
3. Selecciona la opción adicionar.	4. Muestra los campos a llenar.
5. Introduce los datos en los campos. 6. Presiona el botón aceptar.	7. Se validan los datos entrados.

Flujos alternativos

Acción del actor	Respuesta del sistema
	<p>7.1. Si alguno de los datos entrados es incorrecto, se muestra un mensaje de error y se permite entrar los datos nuevamente.</p> <p>7.2. Si el año insertado ya existe se muestra un mensaje y se muestra el listado de los años existentes.</p>
Interfaz I	
	
Flujo Normal de Eventos	
Sección "Modificar Año"	
Acción del actor	Respuesta del sistema
8. El administrador selecciona la opción gestionar año.	9. Se muestra el listado de los años existentes.
10. Selecciona la opción modificar.	11. Muestra los datos del año a modificar.
12. Modifica los datos en los campos. 13. Presiona el botón aceptar.	14. Se validan los datos modificados.
Flujos alternativos	
Acción del actor	Respuesta del sistema
	14.1. Si alguno de los datos entrados es incorrecto, se muestra un mensaje de error y se permite entrar los datos nuevamente.
Interfaz II	

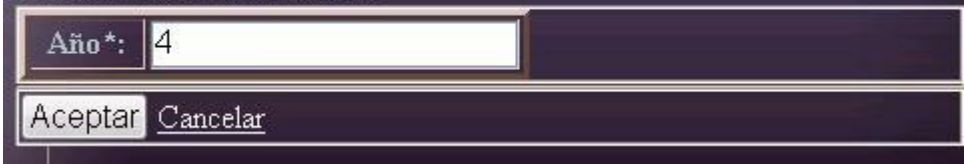
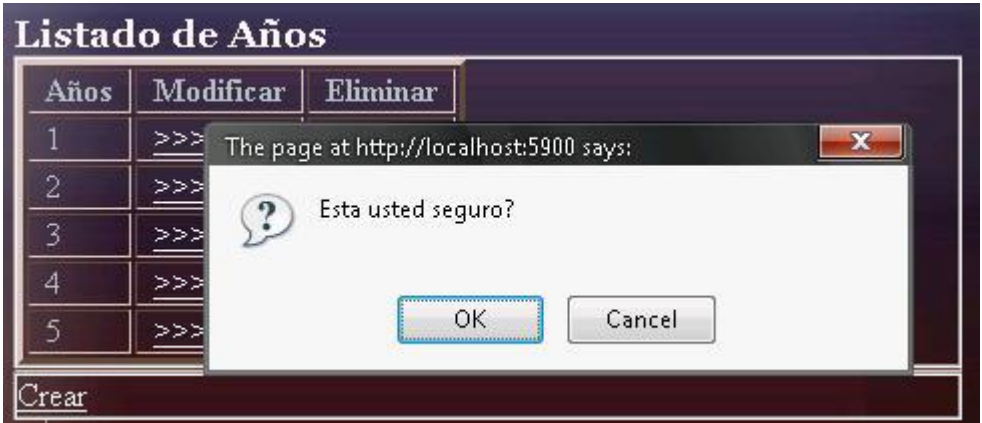
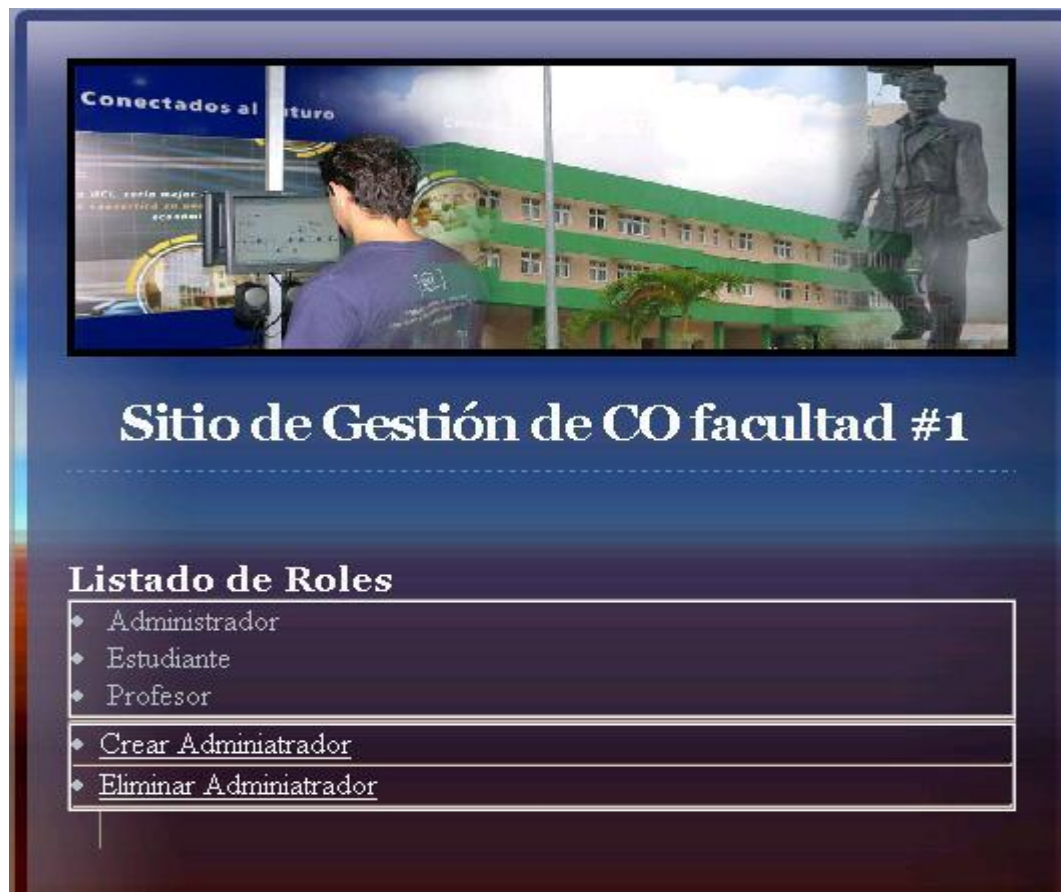
Introduzca los Datos	
	
Flujo Normal de Eventos	
Sección "Eliminar Año"	
Acción del actor	Respuesta del sistema
15. El administrador selecciona la opción gestionar año.	16. Se muestra el listado de los años existentes.
17. Selecciona la opción eliminar.	18. Muestra mensaje de confirmación.
19. Presiona el botón aceptar.	20. Se elimina el año seleccionado.
Flujos alternativos	
Acción del actor	Respuesta del sistema
19.1. Presiona el botón cancelar.	20. Se vuelve a mostrar el listado de los Años existentes.
Interfaz III	
	

Tabla 13: Descripción textual del CUS Gestionar Administrador.


CU-6	Gestionar Administrador
Actores	Administrador

Propósito: Permitir al administrador gestionar los administradores de los CO del sistema.	
Resumen: El caso de uso se inicia cuando el administrador del sistema desea adicionar o eliminar un administrador. El caso de uso termina cuando se realiza algunas de estas acciones.	
Referencias	(5)

Interfaz



Flujo Normal de Eventos	
Sección "Adicionar Administrador"	
Acción del actor	Respuesta del sistema
1. El administrador selecciona la opción gestionar administrador.	2. Se muestran las opciones que posee.
3. Selecciona la opción adicionar.	4. Muestra los campos a llenar.
5. Introduce los datos en los campos.	7. Se validan los datos entrados.

6. Presiona el botón aceptar.	
Flujos alternativos	
Acción del actor	Respuesta del sistema
	7.1. Si alguno de los datos entrados es incorrecto, se muestra un mensaje de error y se permite entrar los datos nuevamente. 7.2. Si el administrador insertado ya existe se muestra un mensaje de error.
Interfaz I	
	
Flujo Normal de Eventos	
Sección "Eliminar Administrador"	
Acción del actor	Respuesta del sistema
8. El administrador selecciona la opción gestionar administrador.	9. Se muestran las opciones.
10. Selecciona la opción eliminar.	11. Muestra los administradores existentes.
12. Selecciona el/los administrador/es.	13. Muestra un mensaje confirmación.
14. Presiona el botón aceptar.	15. Se elimina el administrador seleccionado.
Flujos alternativos	
Acción del actor	Respuesta del sistema
19.1. Presiona el botón cancelar.	20. Se vuelve a mostrar el listado de los Roles existentes.
Interfaz II	

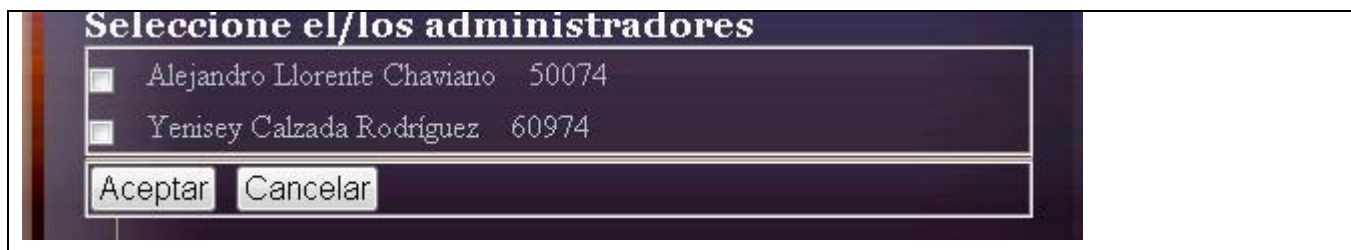


Tabla 14: Descripción textual del CUS Matrícula Forzada.

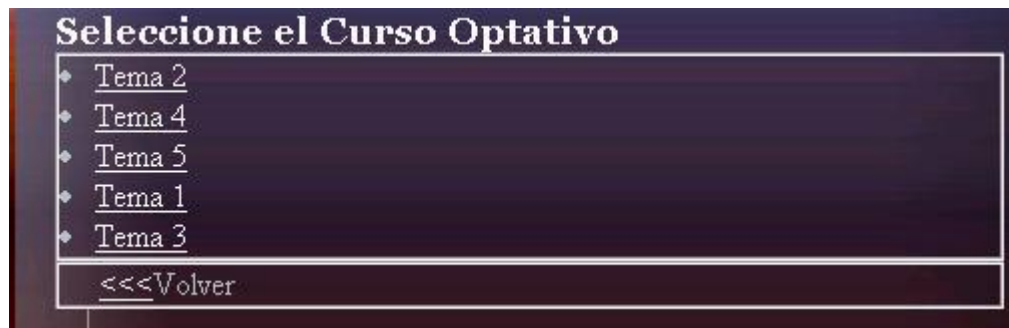
CU-7		Gestionar Matrícula Forzada	
Actores		Administrador	
Propósito: Que el administrador pueda según necesidad de la dirección de la facultad realizar las matriculas de uno o varios estudiantes en uno o varios cursos optativos.			
Resumen: El caso de uso se inicia cuando el administrador desea realizar la matricula o cancelar una previamente realizada de uno o varios estudiantes en uno o varios cursos optativos según necesidad o interés de la dirección de la facultad. El caso de uso concluye cuando se realiza alguna de estas acciones.			
Referencias			
Interfaz			
Flujo Normal de los Eventos			
Sección "Realizar Matrícula Forzada"			
Acción del actor		Respuesta del sistema	
1. Selecciona la opción Matrícula Forzada.		2. Muestra las opciones Realizar Matrícula Forzada y Cancelar Matrícula Forzada.	
3. Selecciona la opción Realizar Matrícula Forzada.		4. Muestra los campos a llenar.	

5. Introduce los datos solicitados.	6. Valida los datos entrados. 7. Si los datos son correctos, realice la matrícula.
-------------------------------------	---

Flujos alternativos

Acción del actor	Respuesta del sistema
	7.1. Si los datos son incorrectos, muestra un mensaje de error, y brinda la opción de corregirlo.

Interfaz I a



Interfaz I b



Sección "Cancelar Matrícula Forzada"

Acción del actor	Respuesta del sistema
1. Selecciona la opción Matrícula Forzada.	2. Muestra las opciones Realizar Matrícula Forzada y Cancelar Matrícula Forzada.
3. Selecciona la opción Cancelar Matrícula Forzada.	4. Muestra los campos a llenar (solapín).
5. Introduce los parámetros solicitados (solapín).	6. Valida los datos entrados (solapín). 7. Si dicho estudiante a realizado alguna

	matrícula, muestra el/los cursos en los que se encuentra matriculado.
8. Selecciona el curso que desea sea cancelada la matrícula.	9. Cancela la matrícula deseada.
Flujos alternativos	
Acción del actor	Respuesta del sistema
	7.1. Si el estudiante no ha realizado ninguna matrícula, muestra un mensaje de error.

Interfaz II a



Interfaz II b



Anexo2. Diagramas de clases del análisis.

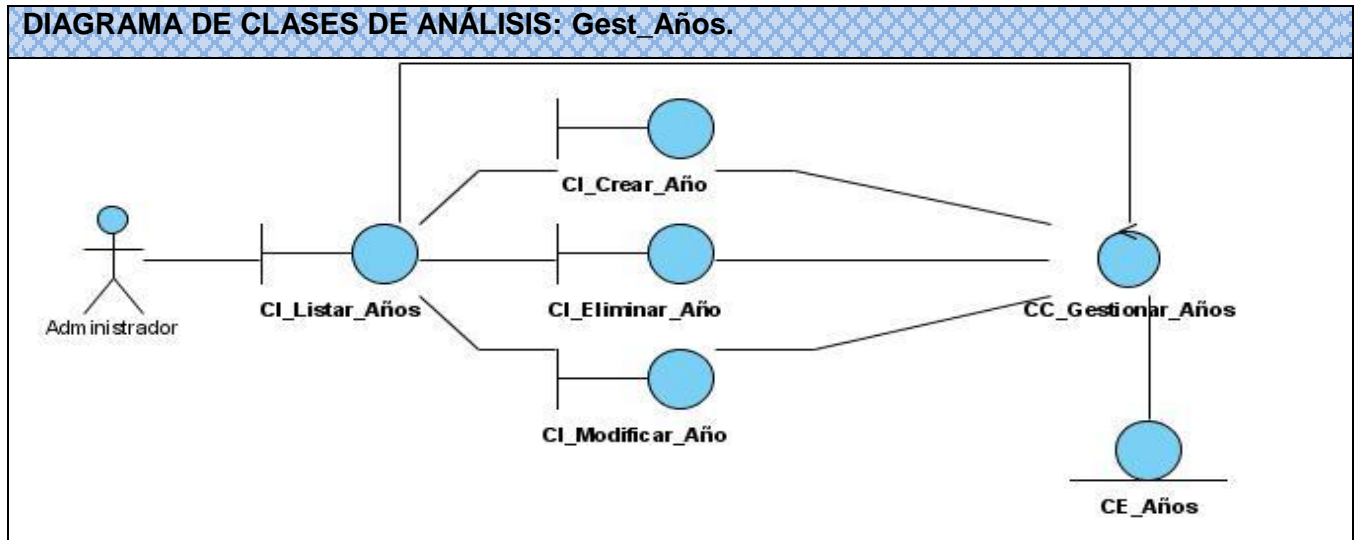


Figura 8: Diagrama de Clases del Análisis Gestionar Años.

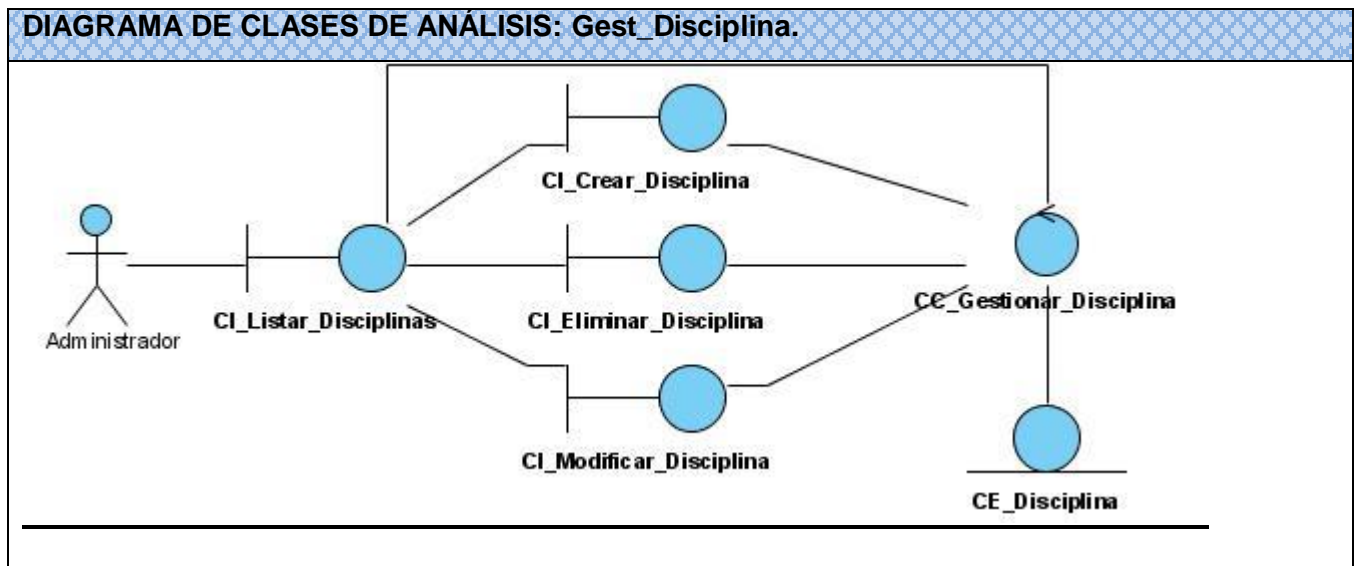


Figura 9: Diagrama de Clases del Análisis Gestionar Disciplina.

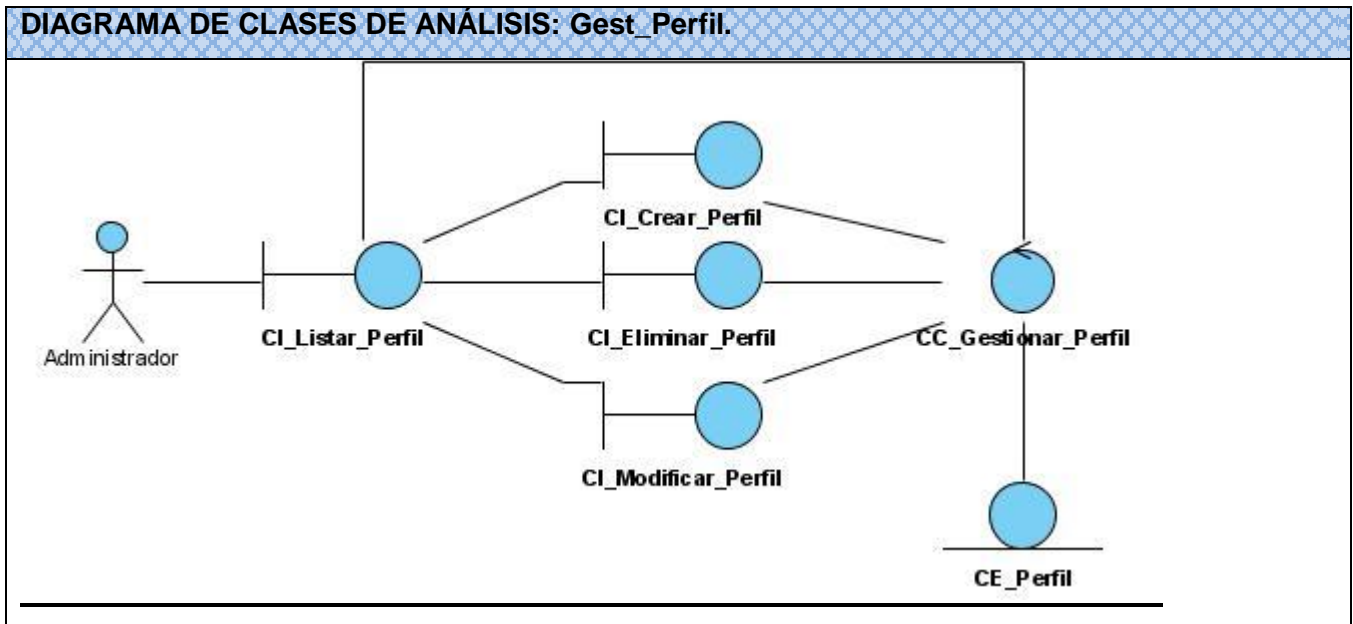


Figura 10: Diagrama de Clases del Análisis Gestionar Perfil.

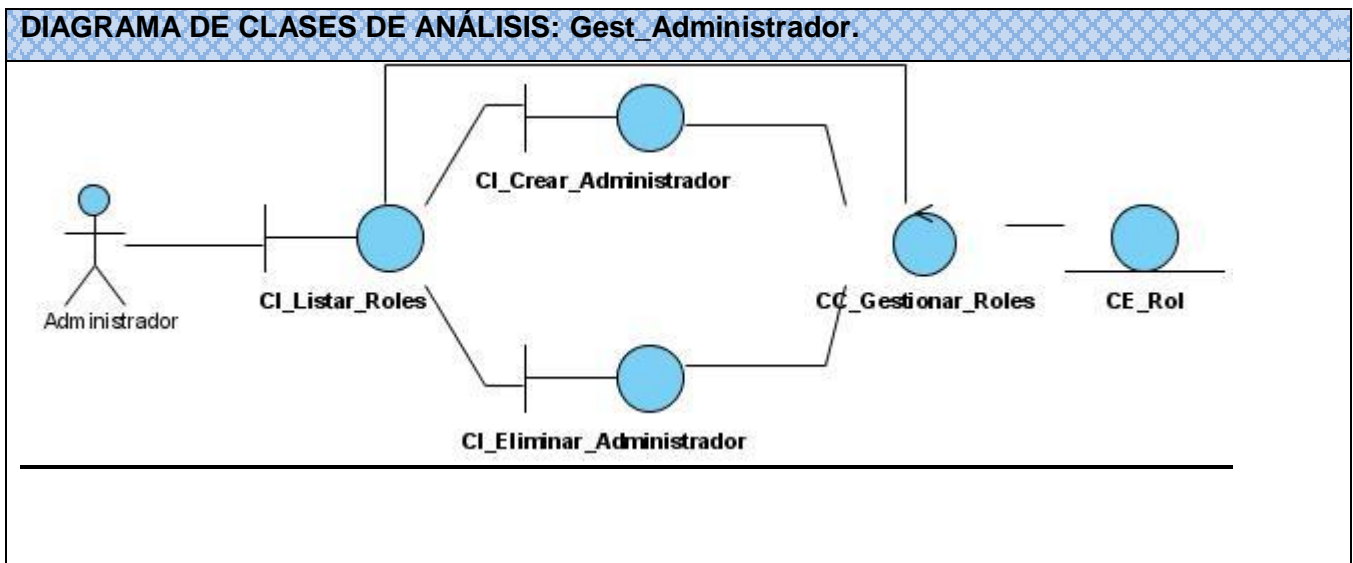


Figura 11: Diagrama de Clases del Análisis Gestionar Administrador.

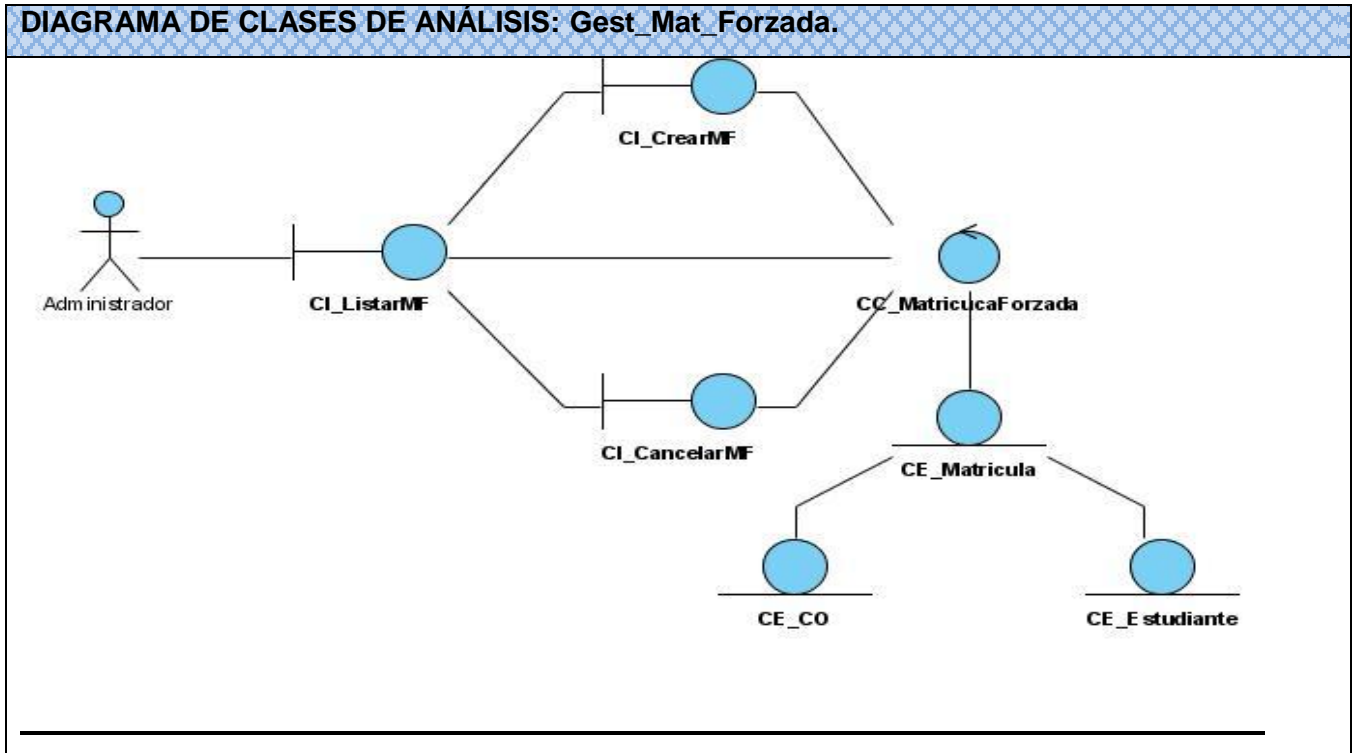


Figura 14: Diagrama de Clases del Análisis Gestionar Matrícula Forzada.

Anexo3. Diagramas de clases del diseño.

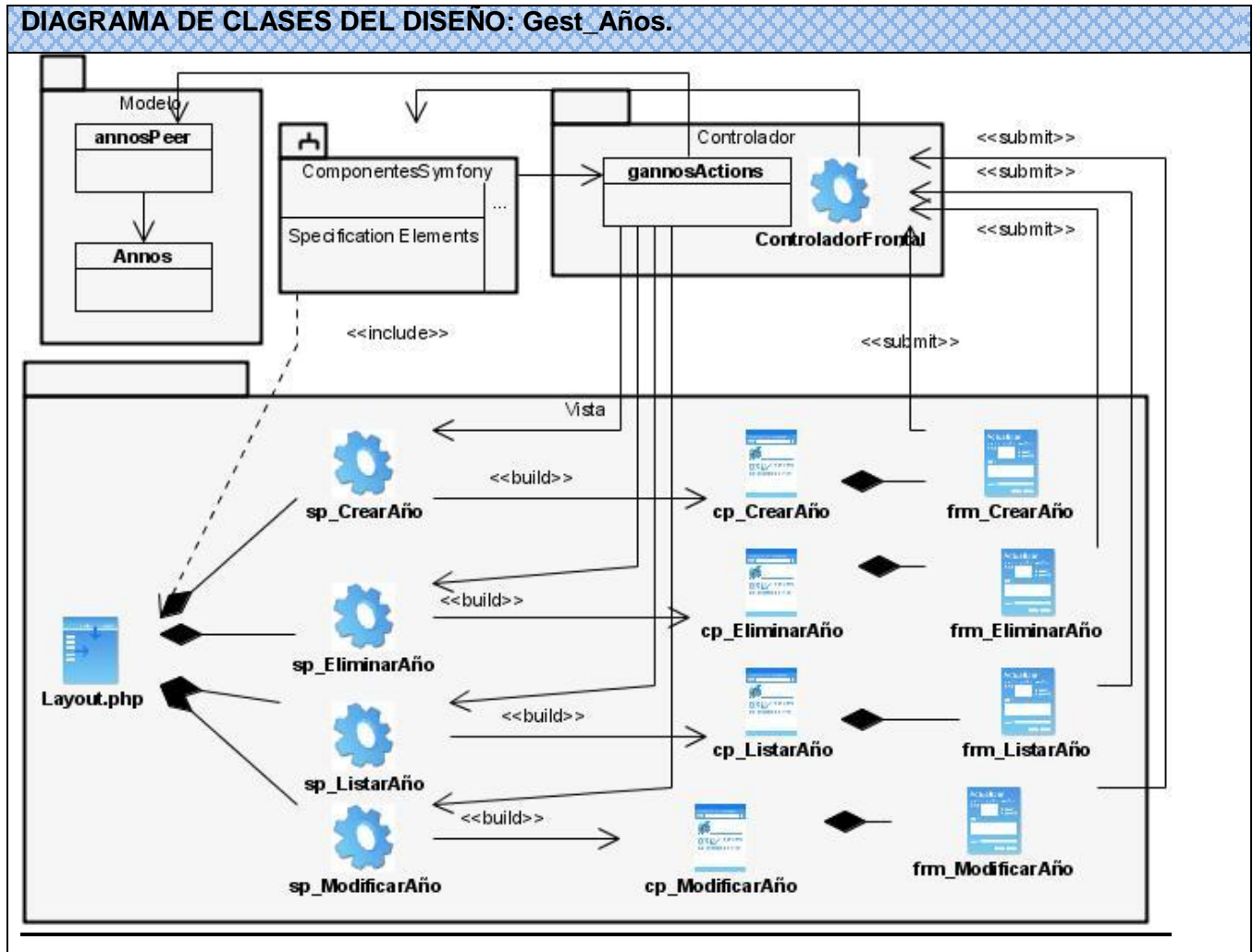


Figura 16: Diagrama de Clases del Diseño Gestionar Años.

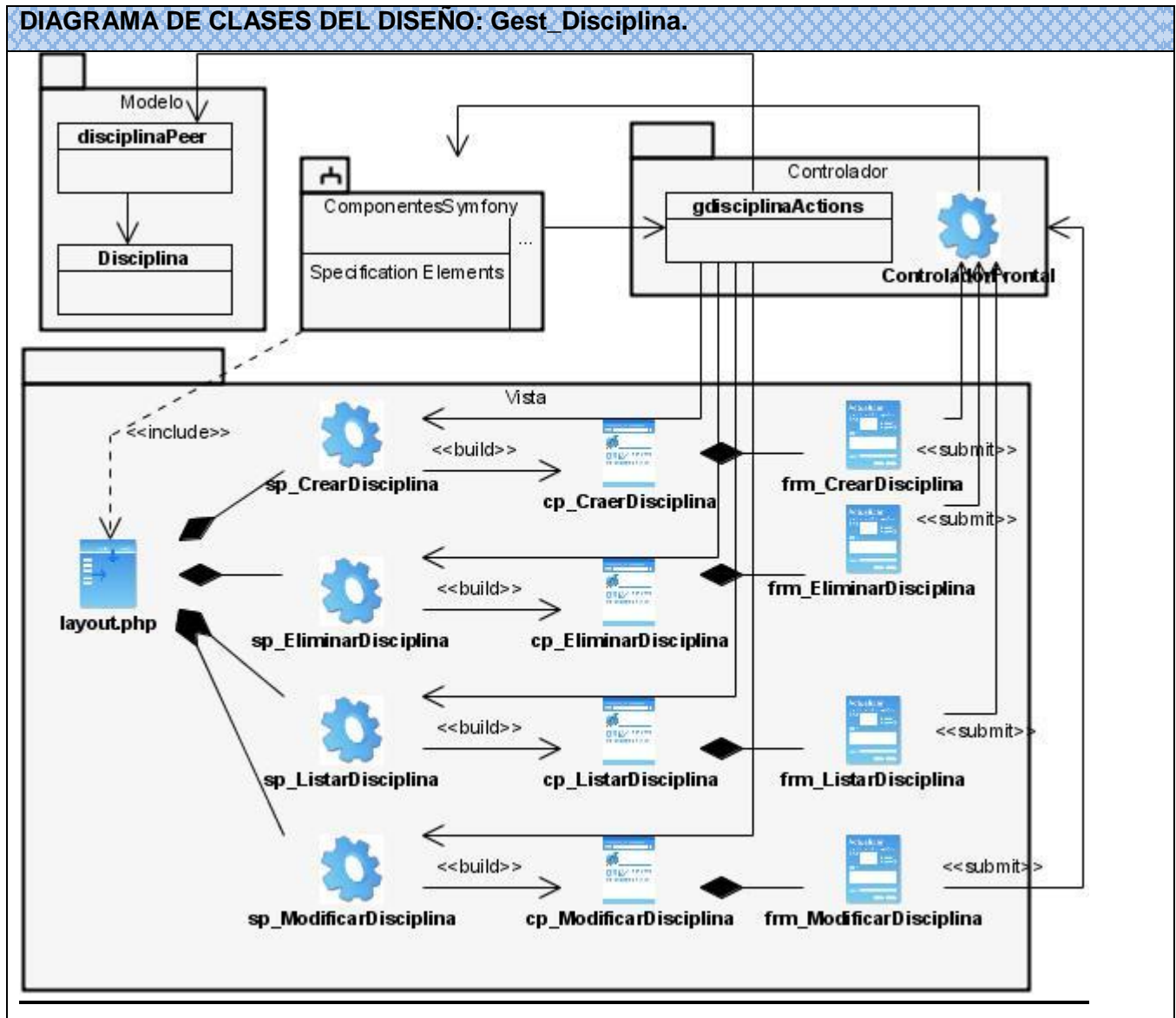


Figura 17: Diagrama de Clases del Diseño Gestionar Disciplina.

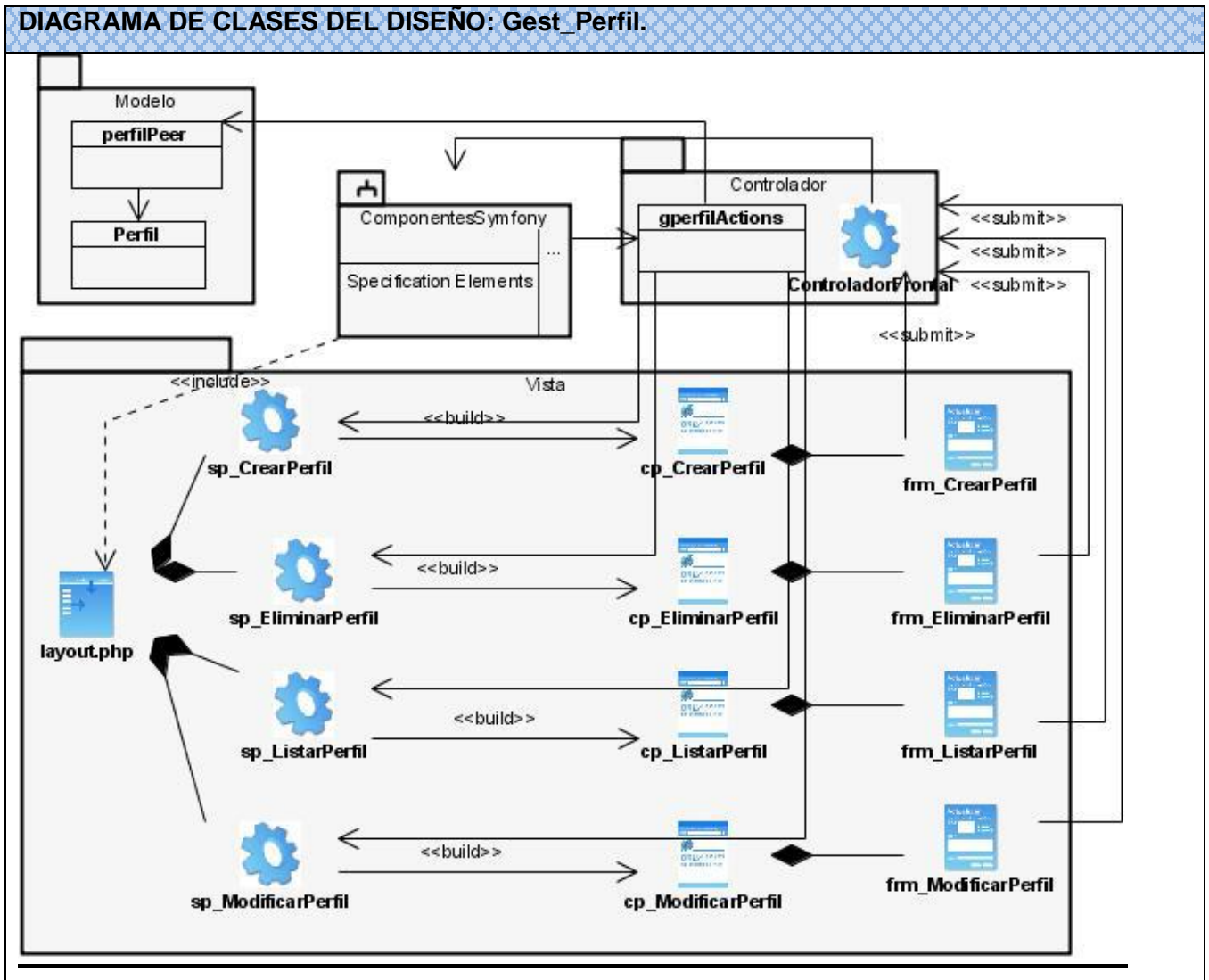


Figura 18: Diagrama de Clases del Diseño Gestionar Perfil.

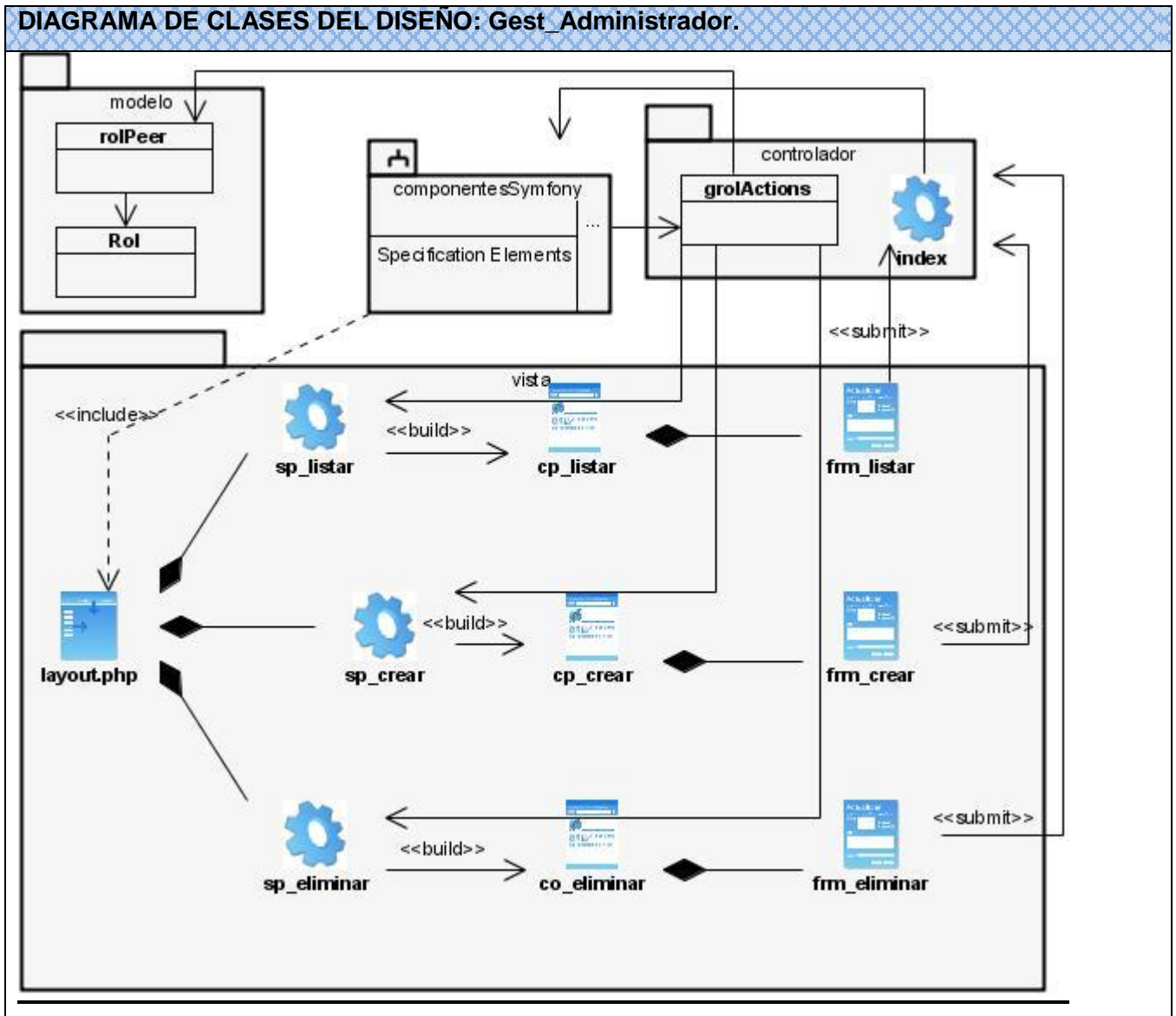


Figura 19: Diagrama de Clases del Diseño Gestionar Administrador.

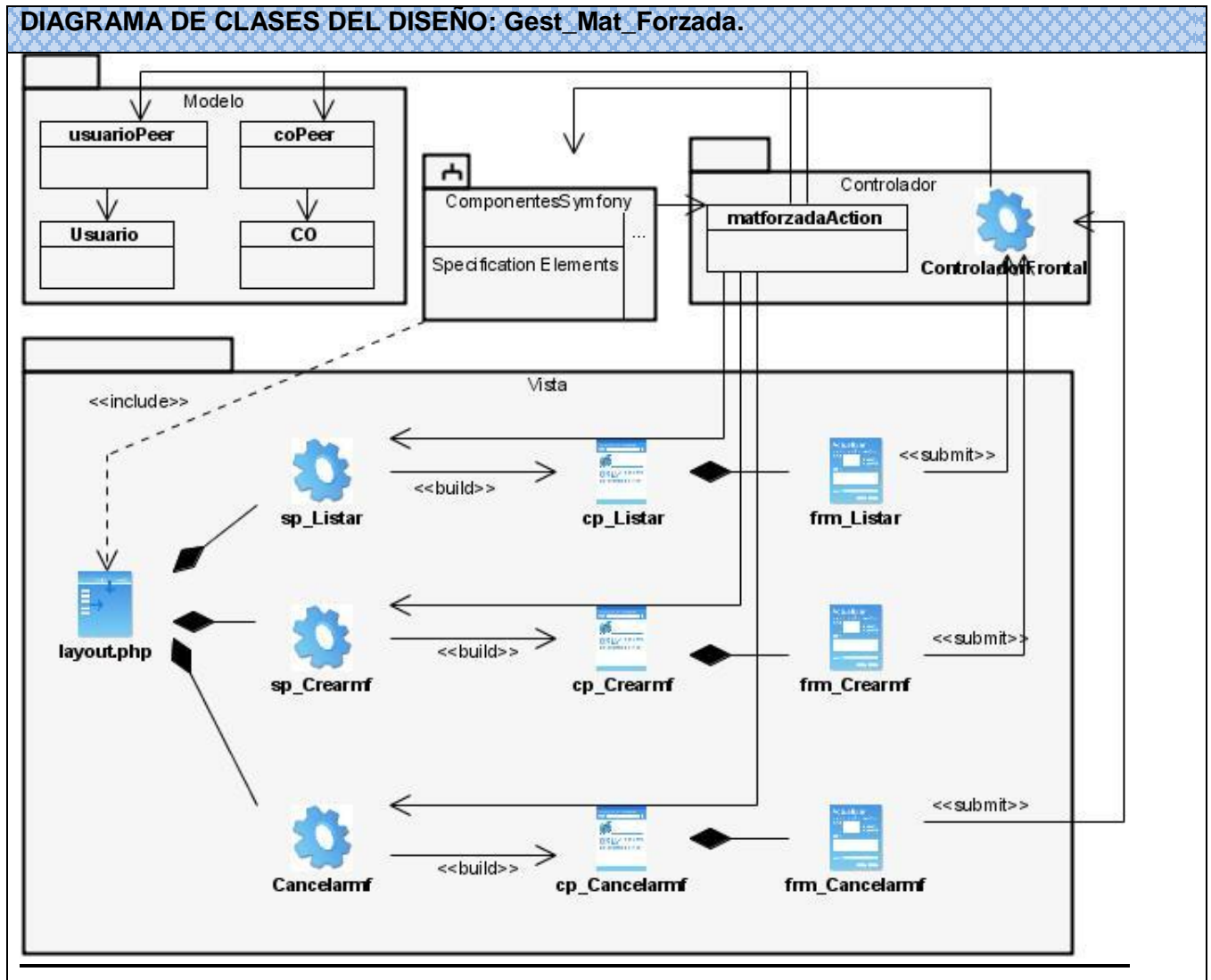


Figura 22: Diagrama de Clases del Diseño Gestionar Matrícula Forzada.

Anexo4. Diagramas de componentes.

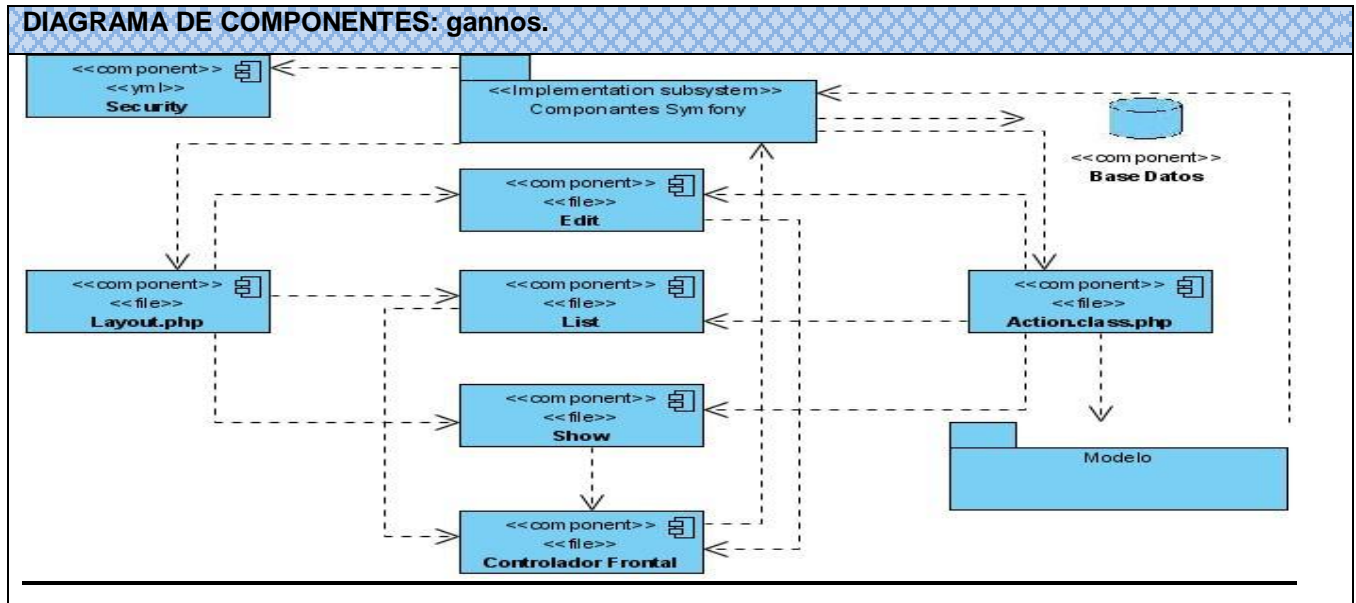


Figura 26: Diagrama de Componentes Gestionar Años.

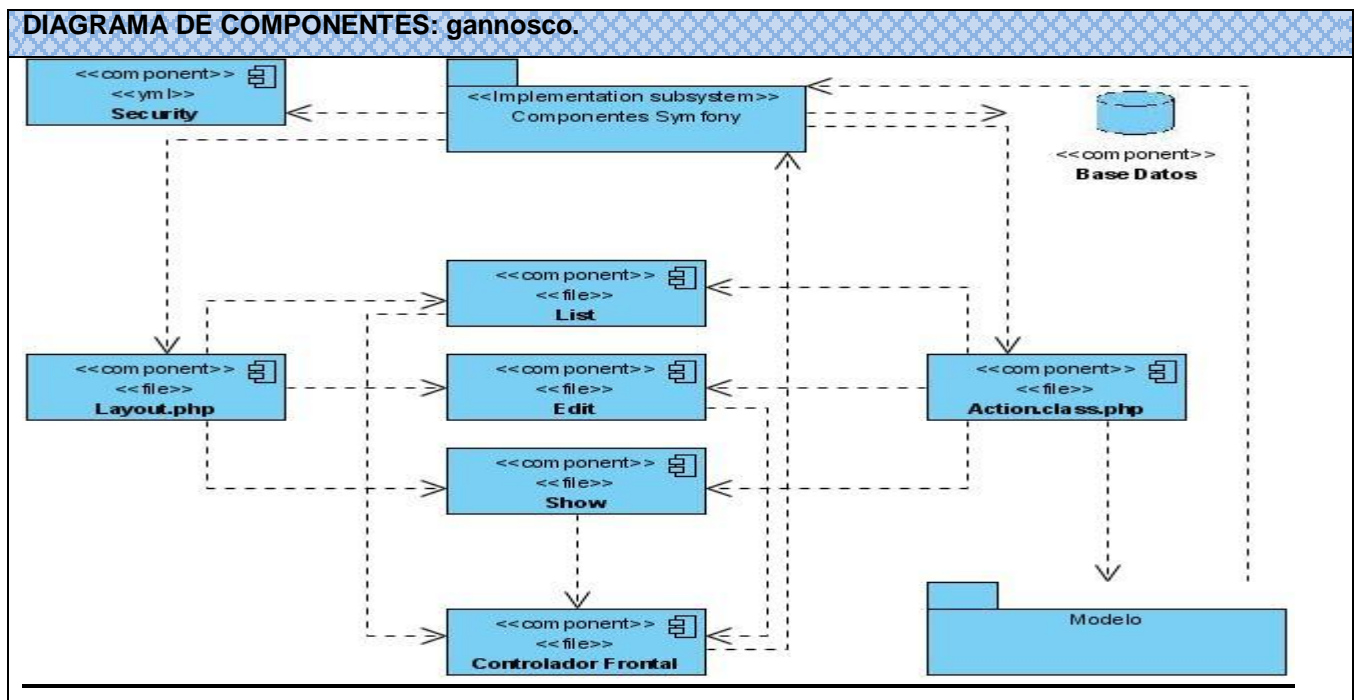


Figura 27: Diagrama de Componentes Gestionar Años-CO.

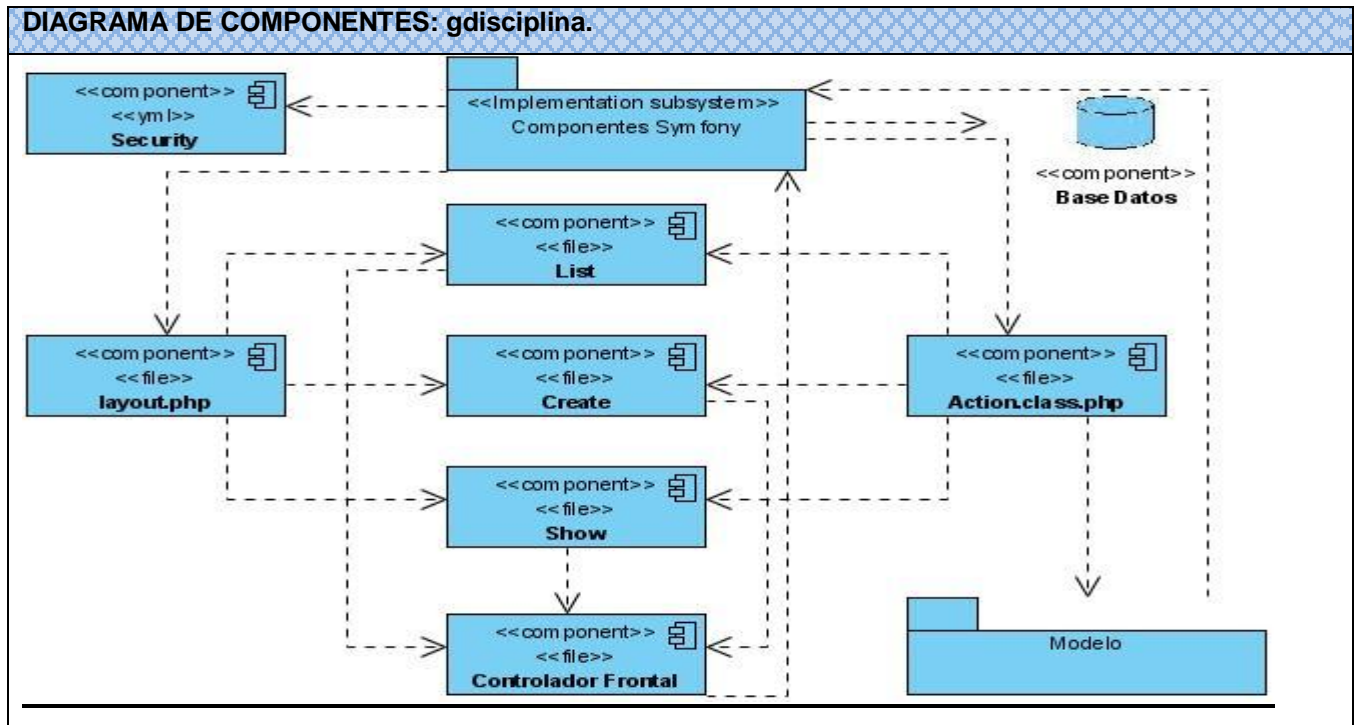


Figura 29: Diagrama de Componentes Gestionar Disciplina.

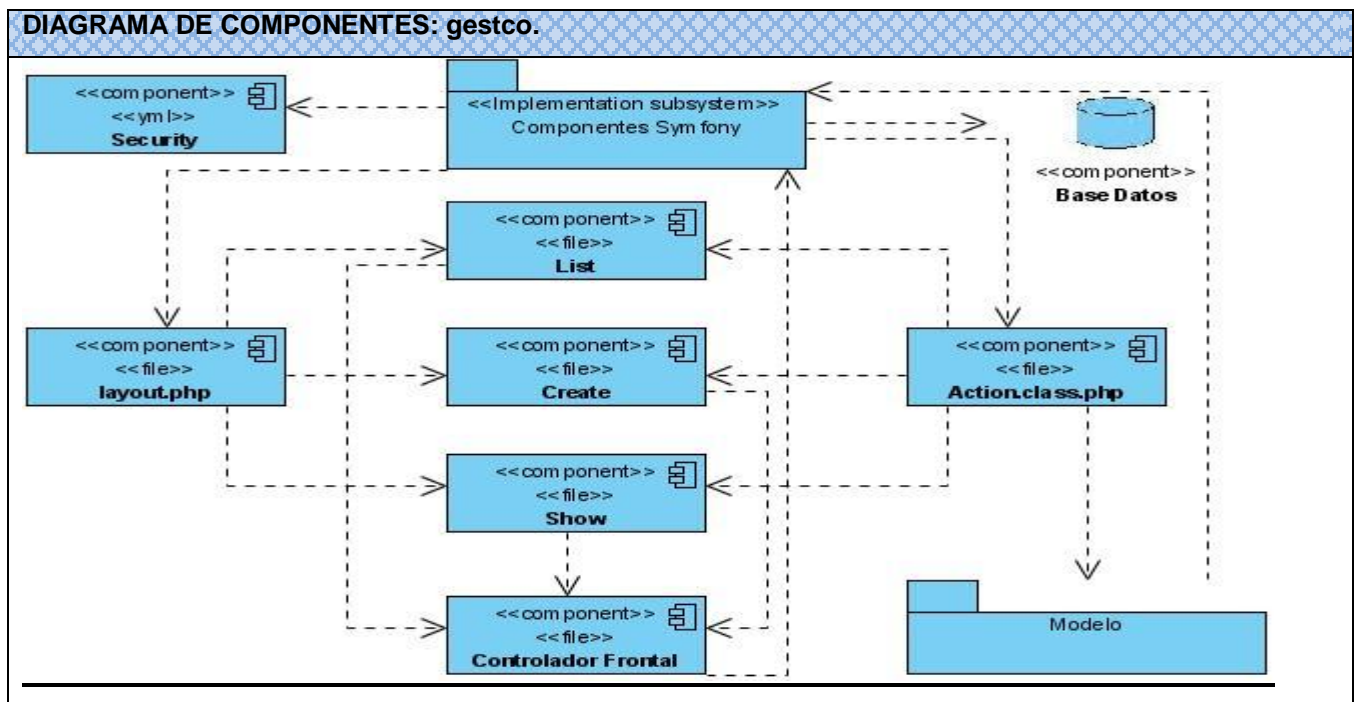


Figura 30: Diagrama de Componentes Gestionar Estudiante-CO.

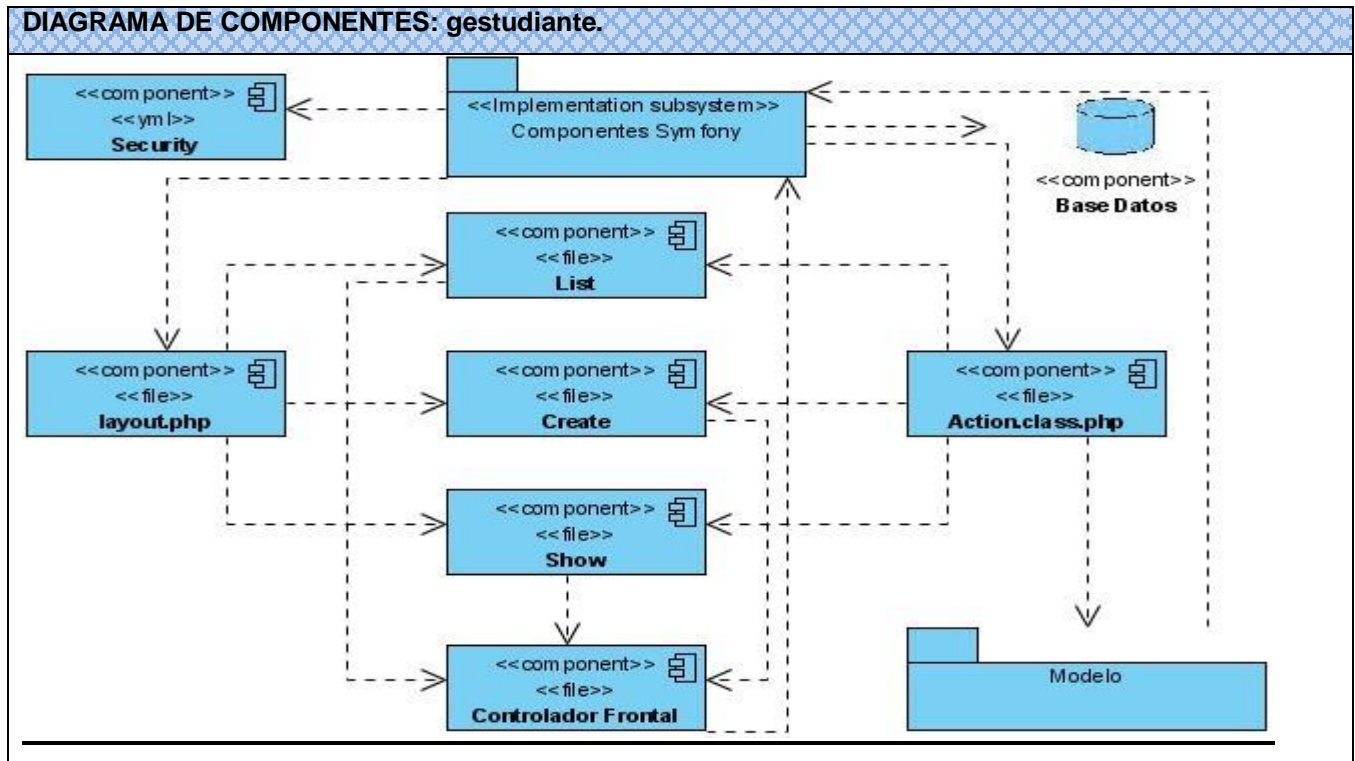


Figura 31: Diagrama de Componentes Gestionar Estudiante.

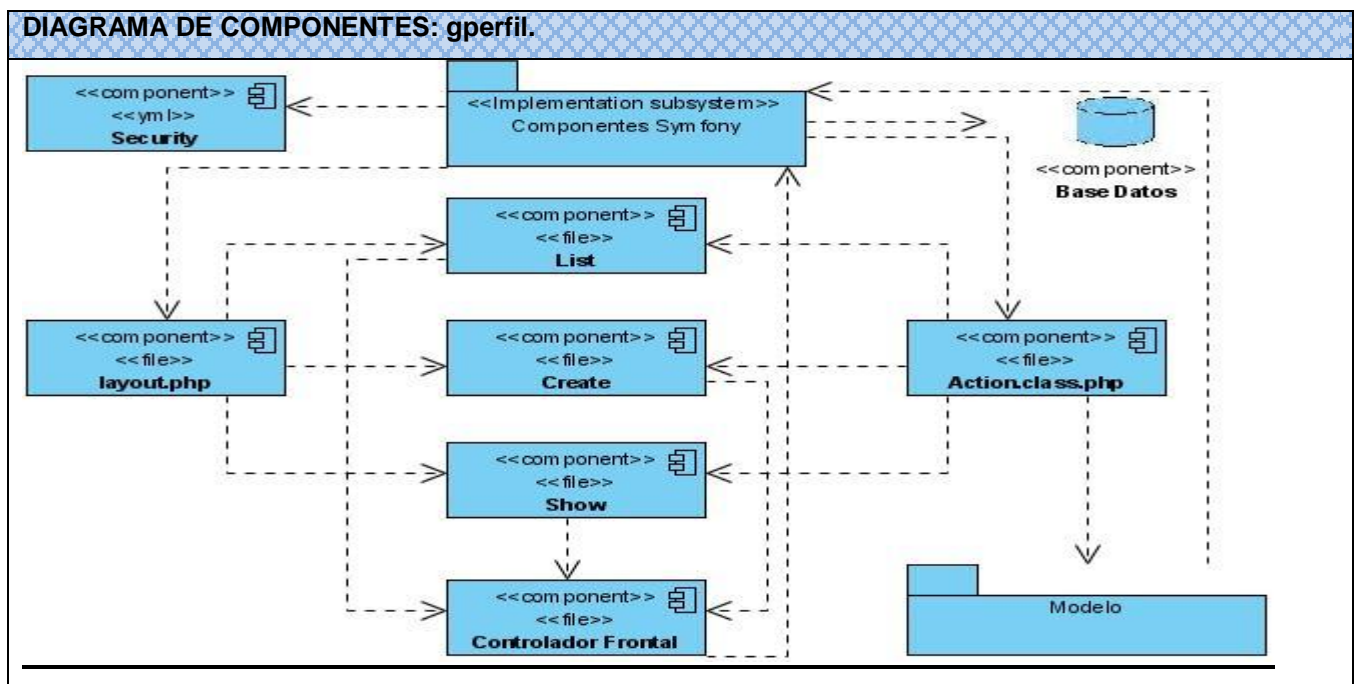


Figura 32: Diagrama de Componentes Gestionar Perfil.

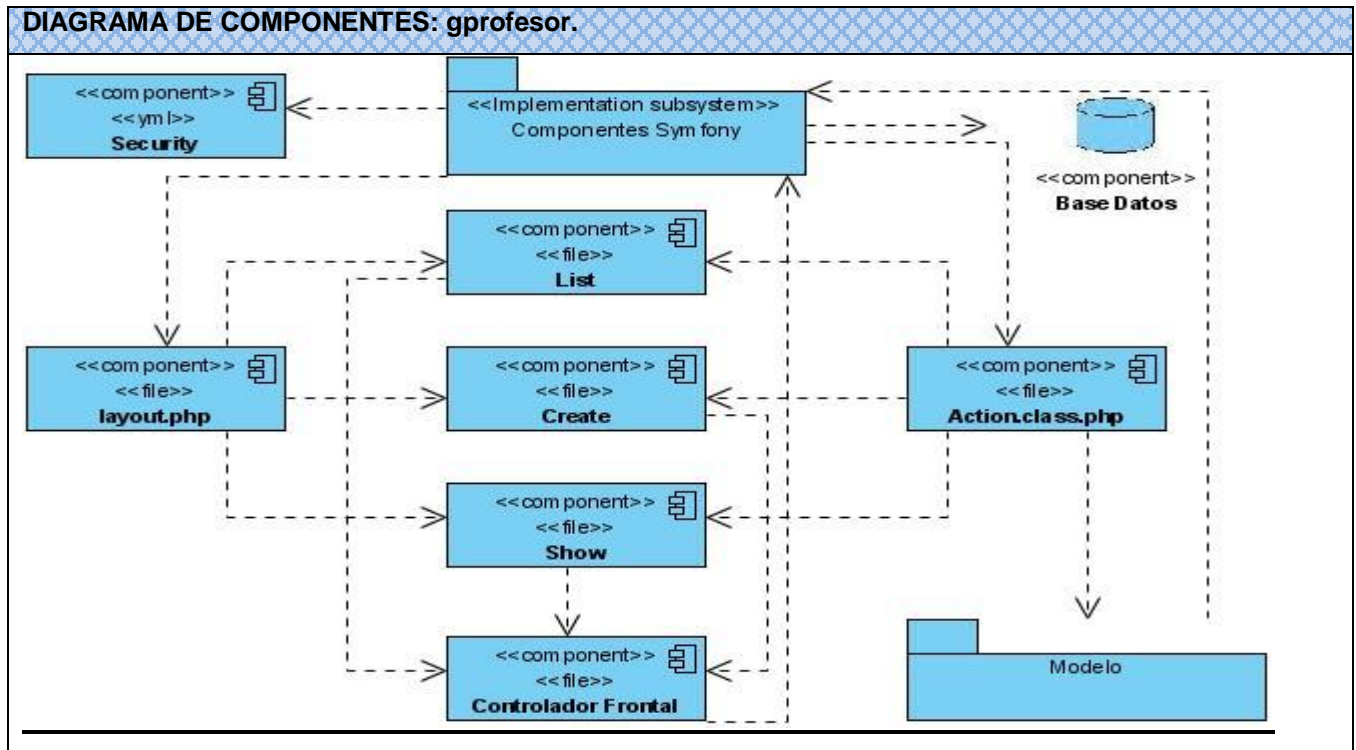


Figura 33: Diagrama de Componentes Gestionar Profesor.

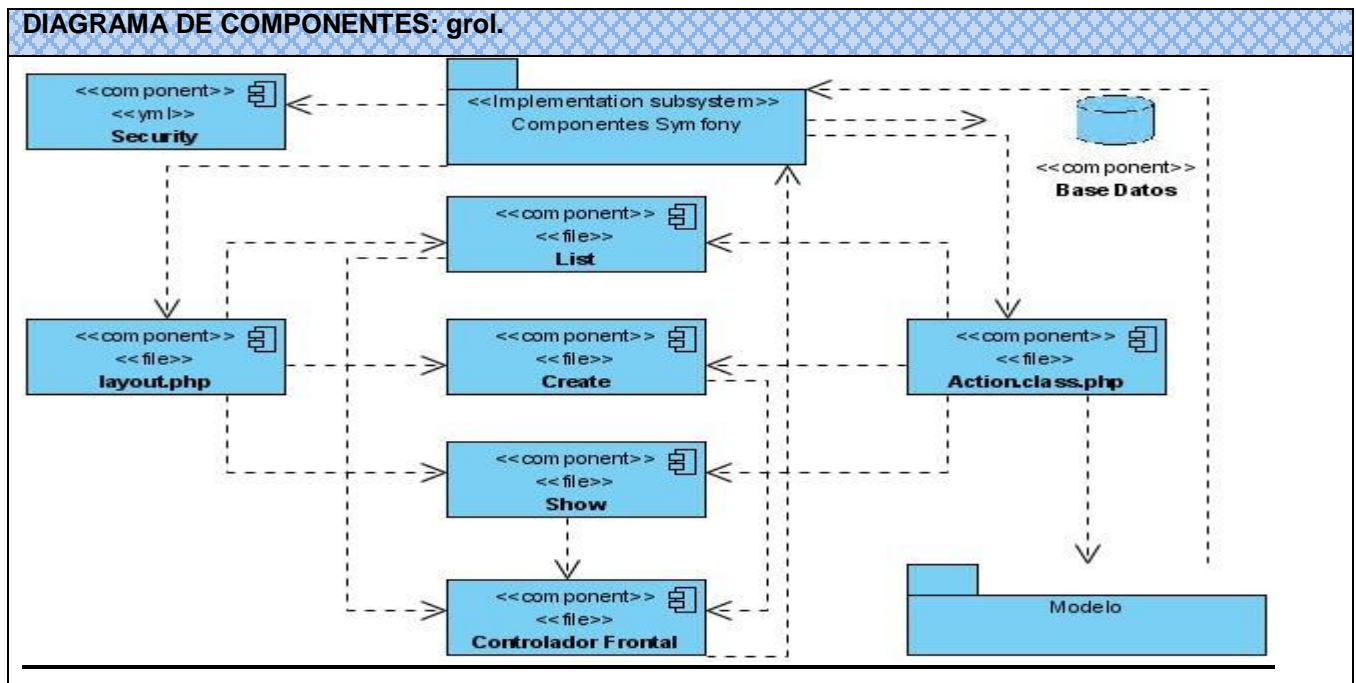


Figura 34: Diagrama de Componentes Gestionar Rol.

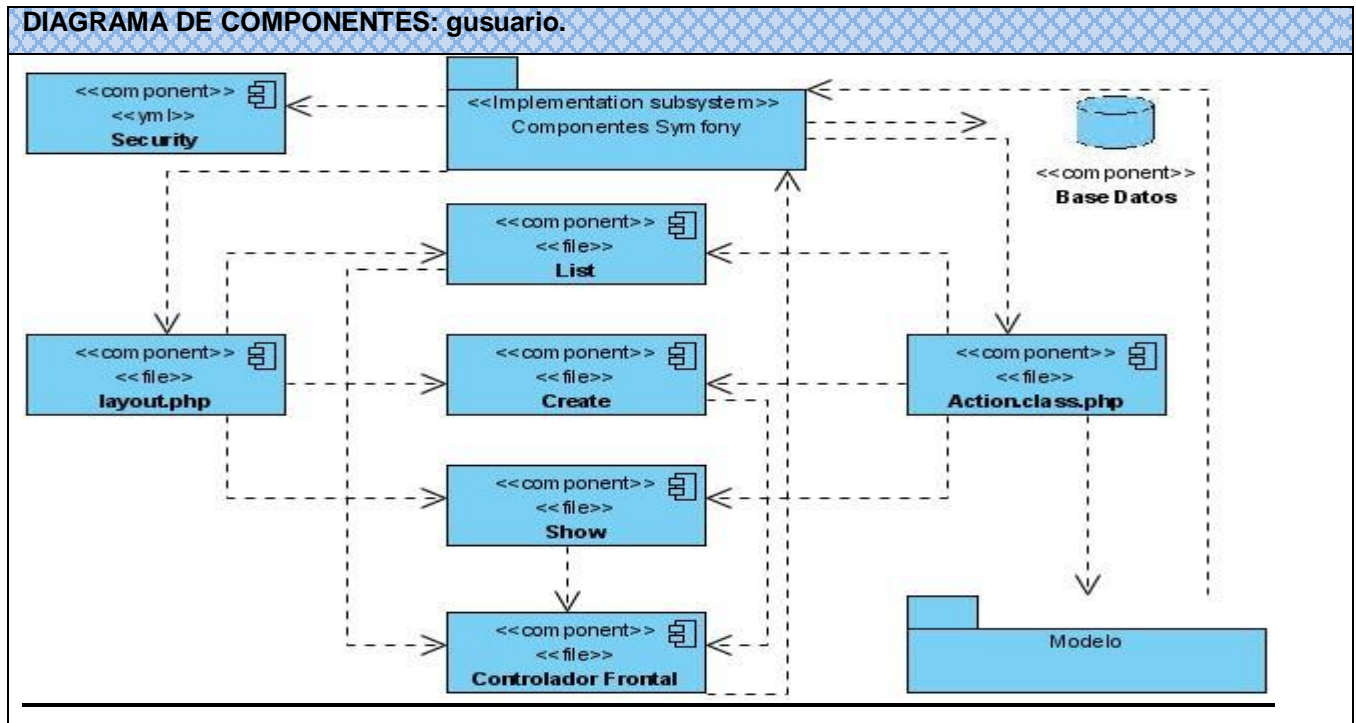


Figura 35: Diagrama de Componentes Gestionar Usuario.

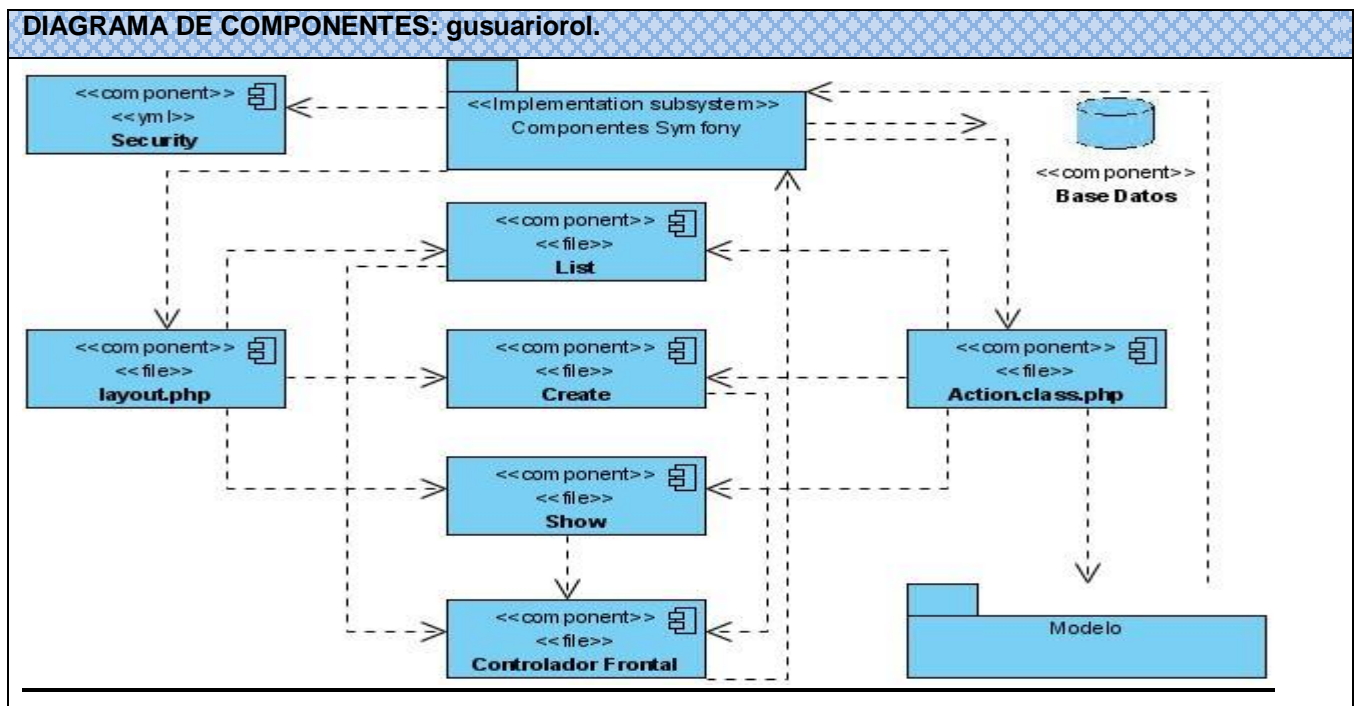


Figura 36: Diagrama de Componentes Gestionar Usuario-Rol.

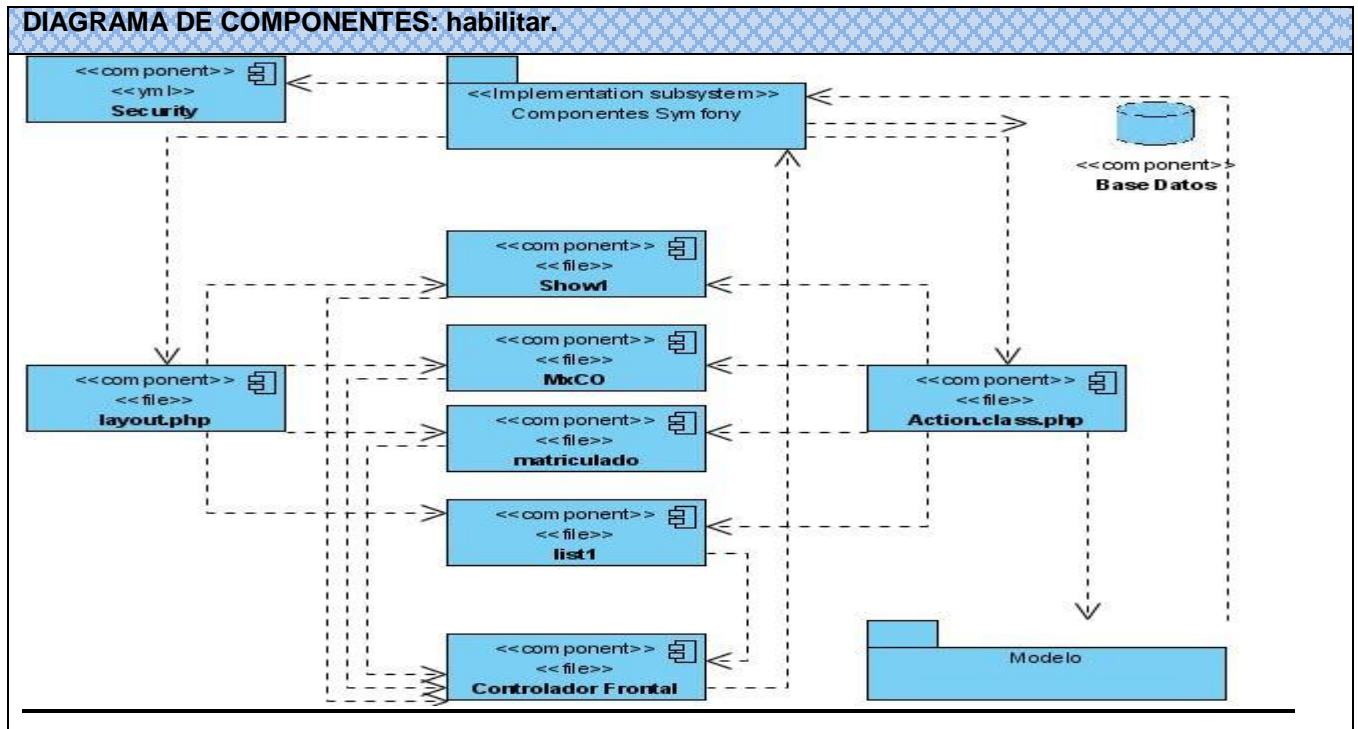


Figura 37: Diagrama de Componentes Habilitar.

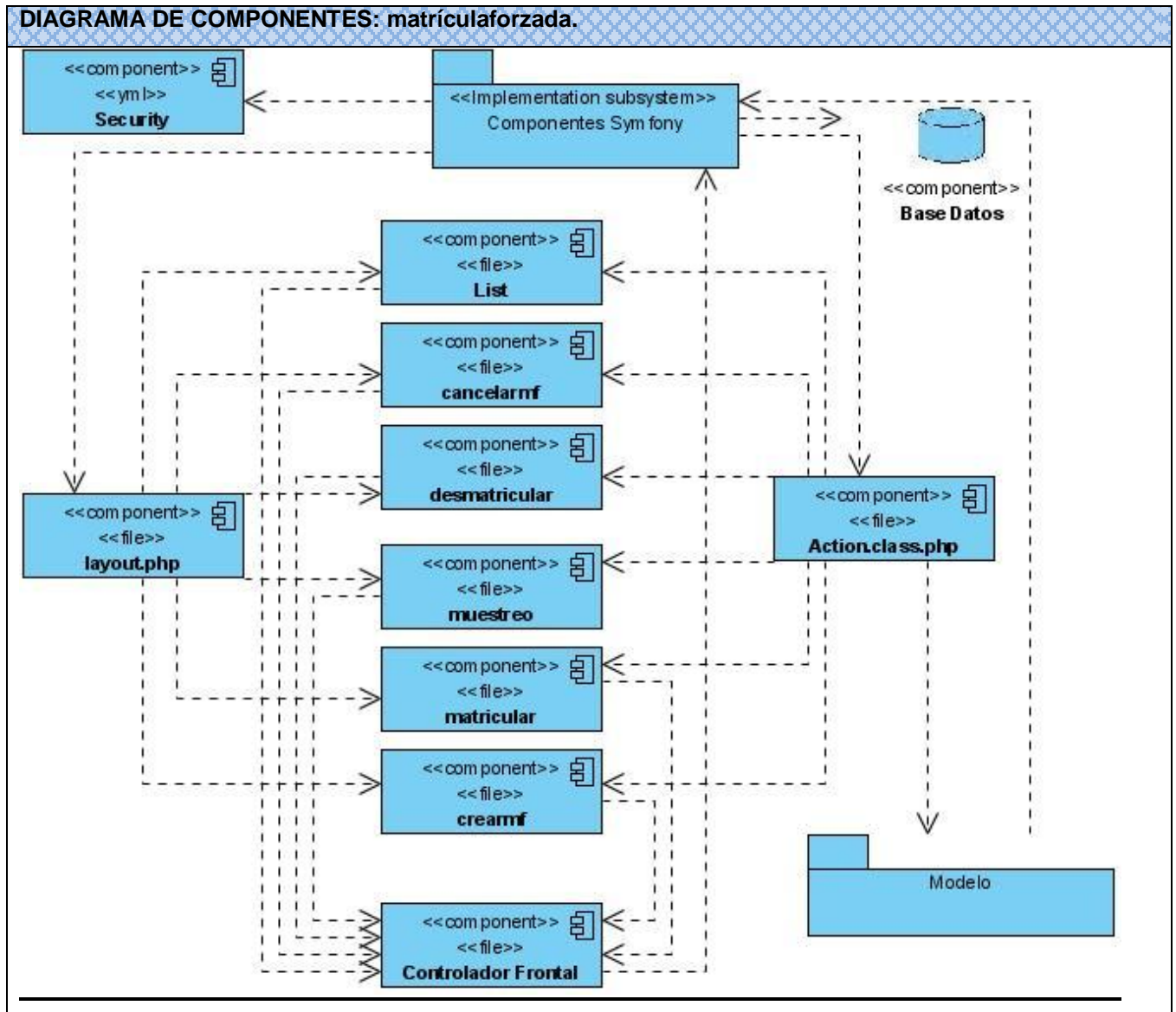


Figura 39: Diagrama de Componentes Matrícula Forzada.