

Universidad de las Ciencias Informáticas

Curso 2008-2009

Facultad 2

Título: Sistema de Gestión de Tesis Facultad 2

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

Autores:

Maité Sosa Veranes.

Vladimir González Guerra.

Tutor:

MSc. Yadira Ruiz Constanten.

Cotutor:

Ing. Devis Ricardo Álvarez Mendoza

Ciudad de la Habana, Junio de 2009.

La inteligencia consiste no sólo en el conocimiento, sino también en la destreza de aplicar los conocimientos en la práctica.

Aristóteles

DECLARACIÓN DE AUTORÍA

Declaramos ser autores de la presente tesis y reconocemos a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste firmamos la presente a los ____ días del mes de _____ del año _____.

Maité Sosa Veranes

Autora

Vladimir González Guerra

Autor

MSc. Yadira Ruiz Constanten

Tutora

Agradezco a:

Mi madre María Antonia Veranes y a mi abuelita Clara Veranes, porque me han dedicado su vida, porque siempre han confiado en mi y por haberme dado su apoyo en todo momento.

Mi novio Leudys Romero, porque ha sido mi soporte, mi estímulo para seguir adelante, porque nunca me ha fallado, y sobre todo, por su paciencia para entenderme.

Toda mi familia, porque siempre me han puesto en lo más alto, dándome un gran voto de confianza, en especial a mis padrinos en La Habana: mi tía Mayuya, Jose y Orbi, por haberme acogido durante estos cinco cursos, dándome su ayuda y cariño.

Todas las personas maravillosas que he conocido durante estos cinco cursos, saber que han estado ahí me hace entender la parte linda de la vida. Gracias a ustedes: a Kiri (porque me entiende como nadie), a Doris (por ayudarme y quererme), a Marcel (porque me ha sido de gran ayuda y nunca me ha dado la espalda), a Rey y al Zuma (por hacerme sentir querida), a Vladimir (por ser tan buen amigo) y a los muchachos desde el 2101 hasta el 2502, porque hemos sido una gran familia, nunca los voy a olvidar, son muy especiales.

Mis tutores Yadira y Deivis porque realmente fueron de gran ayuda para la realización de este trabajo y porque siempre nos dieron su apoyo.

Todas las personas que desde Bayamo siempre me ayudaron y me alentaron para que siguiera adelante: a Maylen, Carmen, Jose, Yusel, mis vecinos Lidia, Ramón, Lily, Sello, Elisa, todos me impulsaron siempre hacia arriba, los quiero mucho.

La Revolución, por darme la oportunidad de estudiar y de hacerme una ingeniera.

Maité

Agradezco a:

Mis padres Caridad Guerra y Eugenio González, que con esfuerzo y desvelo han dedicado sus vidas para cultivarme de valores humanos y forjarme como un hombre de bien. Ellos que han sentado sus esperanzas en verme convertido en profesional y nunca han dudado de mis capacidades y espíritu de lucha.

Todos aquellos que me han acompañado durante estos 5 años, ellos que han sido mi segunda familia. Los que con su sinceridad me han ayudado a crecer y mejorar humanamente, que han ganado un espacio especial en mi corazón y que saben que siempre haré lo posible para que estén orgullosos de mí.

Mi novia Yamilé Rengifo Hardy (mi Tata), por haber tenido tanta paciencia conmigo y apoyarme, y no por hacer distinción entre mis amigos, pero especialmente quiero mencionar a quienes han llegado a ser verdaderos hermanos míos: Maité Sosa Veranes (mi insoportable pero queridísima compañera de tesis y Jefecita), Jorge Javier García Ruiz (aunque sea un loco, pero es mi hermano) y a sus padres, y a Lilibet Sosa Valdés (mi rubia loquita).

De forma general agradezco a todos mis familiares que de una forma u otra me han brindado su apoyo o han mostrado interés y preocupación por mí, así como a las amistades que de igual forma me han ayudado.

Agradezco a Fidel y a la Revolución por hacer posibles los sueños de tantas personas, lograr que la superación profesional esté al alcance de todos y dar la oportunidad a todos de probarnos y demostrar nuestras potencialidades.

Vladimir

DEDICATORIA

A mi madre y a mi abuela, porque son mi espíritu y mi fuerza...

Maité

A mis padres y a Fidel, porque a ellos les debo todo lo que soy...

Vladimir

Resumen

La Universidad de las Ciencias Informáticas tiene entre sus principales metas apoyar el proyecto de informatización del país, así como desarrollar la industria del software para contribuir al desarrollo económico de mismo. Para ello se ha iniciado un proceso de automatización de los servicios y actividades que en ella se efectúan. Esto requiere apoyar y soportar el tratamiento de la información que se maneja, para lo cual se hace necesario llevar un control estricto y eficiente de los documentos, de modo que se pueda garantizar la conservación de los datos y el acceso a estos en el momento que se precise.

El objetivo central de este trabajo es desarrollar una aplicación web que permita optimizar el trabajo y la documentación referente al Proceso de Tesis de Grado en la Facultad 2, brindando la posibilidad de llevar un control durante el transcurso de cada período sobre cada una de las tesis y otros eventos que se efectúan como parte del proceso, el cual se ejecuta hoy de forma manual, lo que influye negativamente en la calidad y efectividad del mismo.

El sistema brinda una amplia información sobre todo el tema referente a las tesis de grado, garantiza la centralización de los documentos generados entorno a este proceso, permite publicar y asignar los temas de tesis propuestos a los estudiantes de quinto año y mantener un seguimiento de la evaluación y evolución de las mismas.

Esta aplicación es independiente del sistema operativo donde se ejecute y presenta un requerimiento de hardware mínimo. Cuenta con una interfaz Web amigable y fácil de usar permitiendo así que un gran número de usuarios puedan disfrutar de sus servicios.

Una vez desarrollada la solución propuesta se dispondrá de un medio eficiente para facilitar el trabajo del personal encargado de realizar la gestión de las tesis de la Facultad 2.

ABSTRACT

The University of Informatics Science has among its main goals to achieve the computerization of our country and develop the software industry to contribute to its economic development. That's why it has engaged in an automation process of services and activities that take place there. This requires support and withstand the processing of information that is handled, which makes it necessary to keep a strict and efficient control of documents, so as to ensure preservation of data and access to these at a time required.

The objective of this work is to develop a web application that allows to optimize the work and documentation regarding the process of thesis at the Faculty 2, offering the possibility of a control during each period on every thesis and other events developed as part of the process, which today is run by hand, which negatively affects the quality and effectiveness.

This system provides extensive information about the whole issue regarding the thesis, it guarantees centralization of all documents generated about this process, and it allows to publish and assign proposed thesis themes to students and to keep a follow-up of the evaluation and evolution of thesis.

This application is independent of the operating system running and has a minimum hardware requirement. It has a Web-friendly interface which is easy to use so that more users can enjoy their services.

After the development of the proposed solution we will have an efficient mean to facilitate the work.

Índice

Introducción	17
Capítulo 1. Fundamentación Teórica.....	22
1.1 Introducción	22
1.2 Estado del Arte	22
1.2.1 Sistemas de Gestión	22
1.2.2 Sistemas de Gestión de Tesis	25
1.3 Justificación de la Propuesta.....	30
1.3.1 Aplicaciones Web.....	30
1.3.2 Modelo Cliente – Servidor	31
1.3.2.1 Tecnologías del lado del cliente.....	33
1.3.2.2 Tecnologías del lado del servidor.....	37
1.3.3 Web 2.0	45
1.3.4 Framework.....	46
1.3.4.1 CodeIgniter	47
1.4 Fundamentación de la Metodología	49
1.4.1 Herramientas Case	52
1.5 Conclusiones	54
Capítulo 2. Características del sistema.	55
2.1 Introducción	55
2.2 Objeto de estudio	55
2.2.1 Problema y situación problemática	55
2.2.2 Objeto de automatización.	55
2.2.3 Información que se maneja.	56
2.3 Propuesta del sistema.....	56

2.4	Modelo de negocio	56
2.4.1	Descripción General del Negocio	57
2.4.2	Actores del Negocio	57
2.4.3	Trabajadores del Negocio	58
2.4.4	Diagrama de Casos de Uso del Negocio.....	58
2.4.5	Descripción textual de los Casos de Uso del Negocio. Ver anexo 1.....	59
2.4.6	Diagramas de Actividades. Ver anexo 2.....	59
1.4.7	Modelo de Objetos. Ver anexo 3.....	59
1.5	Relación de requerimientos	59
2.5.1	Requerimientos funcionales	59
2.5.2	Requerimientos no funcionales	63
2.6	Modelo de Casos de Uso del Sistema	65
2.6.1	Definición de los actores del sistema a automatizar	65
2.6.2	Diagrama de casos de uso del sistema a automatizar	67
2.6.3	Patrones de casos de uso.....	67
2.6.4	Descripción textual de los Casos de Uso del Sistema. Ver anexo 4	68
2.7	Conclusiones	68
Capítulo 3. Análisis y diseño del sistema		69
3.1	Introducción	69
3.2	Definición de diagrama de clases	69
3.3	Análisis del sistema	69
3.3.1	Modelo de Análisis	69
3.3.2	Diagramas de clases del análisis.....	70
3.4	Diseño del sistema	83
3.4.1	Modelo del Diseño	83
3.4.2	Diagramas de clases del diseño.....	83
3.4.3	Patrones de diseño.....	97
3.4.4	Diagramas de interacción	98

3.4.4.1	Diagramas de Secuencia.....	98
3.5	Diseño de la Base de Datos	98
3.5.1	Modelo lógico de datos. Diagrama de clases persistentes	99
3.5.2	Modelo físico de datos. Modelo de datos	101
3.6	Patrón arquitectónico. Modelo-Vista-Controlador	102
3.7	Tratamiento de errores.....	103
3.8	Seguridad.....	103
3.9	Conclusiones	104
Capítulo 4. Implementación		105
4.1	Introducción	105
4.2	Diagrama de despliegue.....	105
4.3	Diagrama de componentes	106
4.4	Conclusiones	114
Capítulo 5. Estudio de Factibilidad		115
5.1	Introducción	115
5.2	Estimación por puntos de casos de uso.....	115
5.3	Beneficios tangibles e intangibles	122
5.4	Análisis de costos	123
5.5	Conclusiones	123
Conclusiones Generales.....		124
Recomendaciones.....		125
Referencias bibliográficas		126

Bibliografía.....	129
Anexos.....	134
<i>Anexo 1. Descripción textual de los Casos de Uso del Negocio en formato expandido.....</i>	<i>134</i>
Anexo 2. Diagramas de Actividades.....	139
<i>Anexo 3. Modelo de objetos.....</i>	<i>142</i>
<i>Anexo 4. Descripción textual de los Casos de Uso del Sistema.....</i>	<i>142</i>
<i>Anexo 5. Diagramas de interacción del diseño.....</i>	<i>180</i>
Glosario de Términos.....	188

ÍNDICE DE FIGURAS

FIGURA 1: PANTALLA PRINCIPAL DE LA APLICACIÓN GESTOR DE TESIS	27
FIGURA 2: MODELO CLIENTE - SERVIDOR.....	32
FIGURA 3: DIAGRAMA DE CASOS DE USO DEL NEGOCIO	59
FIGURA 4: DIAGRAMA DE CASOS DE USO DEL SISTEMA	67
FIGURA 5: DIAGRAMA DE CLASES DEL ANÁLISIS DEL CUS “AUTENTICAR USUARIO”	70
FIGURA 6: DIAGRAMA DE CLASES DEL ANÁLISIS DEL CUS “EVALUAR PERFIL”	71
FIGURA 7: DIAGRAMA DE CLASES DEL ANÁLISIS DEL CUS “GESTIONAR PERFIL DE TESIS”	72
FIGURA 8: DIAGRAMA DE CLASES DEL ANÁLISIS DEL CUS “GESTIONAR CORTES DE TESIS”	73
FIGURA 9: DIAGRAMA DE CLASES DEL ANÁLISIS DEL CUS “GESTIONAR DOCUMENTOS DE TESIS”	74
FIGURA 10: DIAGRAMA DE CLASES DEL ANÁLISIS DEL CUS “GESTIONAR TRIBUNAL DE TESIS”	75
FIGURA 11: DIAGRAMA DE CLASES DEL ANÁLISIS DEL CUS “REALIZAR COMENTARIOS”	76
FIGURA 12: DIAGRAMA DE CLASES DEL ANÁLISIS DEL CUS “GESTIONAR PROYECTOS”	77
FIGURA 13: DIAGRAMA DE CLASES DEL ANÁLISIS DEL CUS “GESTIONAR COMENTARIOS”	78
FIGURA 14: DIAGRAMA DE CLASES DEL ANÁLISIS DEL CUS “GESTIONAR USUARIOS”	79
FIGURA 15: DIAGRAMA DE CLASES DEL ANÁLISIS DEL CUS “GESTIONAR PROPUESTAS DE TESIS”	80
FIGURA 16: DIAGRAMA DE CLASES DEL ANÁLISIS DEL CUS “GESTIONAR EVALUACIONES”	81
FIGURA 17: DIAGRAMA DE CLASES DEL ANÁLISIS DEL CUS “REALIZAR BÚSQUEDAS”	82
FIGURA 18: DIAGRAMA DE CLASES DEL DISEÑO DEL CUS “AUTENTICAR USUARIO”	84
FIGURA 19: DIAGRAMA DE CLASES DEL DISEÑO DEL CUS “EVALUAR PERFIL”	85
FIGURA 20: DIAGRAMA DE CLASES DEL DISEÑO DEL CUS “GESTIONAR PERFIL DE TESIS”	86
FIGURA 21: DIAGRAMA DE CLASES DEL DISEÑO DEL CUS “GESTIONAR CORTES DE TESIS”	87
FIGURA 22: DIAGRAMA DE CLASES DEL DISEÑO DEL CUS “GESTIONAR DOCUMENTOS DE TESIS”	88
FIGURA 23: DIAGRAMA DE CLASES DEL DISEÑO DEL CUS “GESTIONAR TRIBUNAL DE TESIS”	89
FIGURA 24: DIAGRAMA DE CLASES DEL DISEÑO DEL CUS “GESTIONAR PROYECTOS”	90
FIGURA 25: DIAGRAMA DE CLASES DEL DISEÑO DEL CUS “REALIZAR COMENTARIOS”	91
FIGURA 26: DIAGRAMA DE CLASES DEL DISEÑO DEL CUS “GESTIONAR COMENTARIOS”	92

FIGURA 27: DIAGRAMA DE CLASES DEL DISEÑO DEL CUS “GESTIONAR USUARIOS”	93
FIGURA 28: DIAGRAMA DE CLASES DEL DISEÑO DEL CUS “GESTIONAR PROPUESTAS”	94
FIGURA 29: DIAGRAMA DE CLASES DEL DISEÑO DEL CUS “GESTIONAR EVALUACIONES”	95
FIGURA 30: DIAGRAMA DE CLASES DEL DISEÑO DEL CUS “REALIZAR REPORTES”	96
FIGURA 31: MODELO LÓGICO DE DATOS	100
FIGURA 32: MODELO FÍSICO DE DATOS	101
FIGURA 33: INTERACCIÓN ENTRE LOS COMPONENTES DEL PATRÓN MVC.....	102
FIGURA 34: DIAGRAMA DE DESPLIEGUE DEL SISTEMA	105
FIGURA 35: DIAGRAMA DE COMPONENTES DEL CU “AUTENTICAR USUARIO”	106
FIGURA 36: DIAGRAMA DE COMPONENTES DEL CU “EVALUAR PERFIL”.....	107
FIGURA 37: DIAGRAMA DE COMPONENTES DEL CU “GESTIONAR CORTES DE TESIS”	108
FIGURA 38: DIAGRAMA DE COMPONENTES DEL CU “GESTIONAR DOCUMENTOS”	109
FIGURA 39: DIAGRAMA DE COMPONENTES DEL CU “GESTIONAR PROPUESTAS DE TESIS”	110
FIGURA 40: DIAGRAMA DE COMPONENTES DEL CU “GESTIONAR PROYECTOS”	111
FIGURA 41: DIAGRAMA DE COMPONENTES DEL CU “GESTIONAR TRIBUNAL DE TESIS”	112
FIGURA 42: DIAGRAMA DE COMPONENTES DEL CU “GESTIONAR USUARIOS”	113
FIGURA 43: DIAGRAMA DE COMPONENTES DEL CU “GESTIONAR PERFIL DE TESIS”	114
FIGURA 44: DIAGRAMA DE ACTIVIDADES DEL CUN “REALIZAR TESIS”	139
FIGURA 45: DIAGRAMA DE ACTIVIDADES DEL CUN “ASIGNAR PERFIL DE TESIS”	140
FIGURA 46: DIAGRAMA DE ACTIVIDADES DEL CUN “REALIZAR CORTE DE TESIS”	141
FIGURA 47: MODELO DE OBJETOS	142

Índice de tablas

TABLA 1: RELACIÓN DE ACTORES DEL NEGOCIO.....	57
TABLA 2: RELACIÓN DE TRABAJADORES DEL NEGOCIO.....	58
TABLA 3: RELACIÓN DE ACTORES DEL SISTEMA.....	66
TABLA 4: FACTOR DE PESO DE LOS ACTORES SIN AJUSTAR (UAW).....	116
TABLA 5: FACTOR DE PESO DE LOS CASOS DE USO SIN AJUSTAR (UUCW).....	117
TABLA 6: FACTOR DE COMPLEJIDAD TÉCNICA (TCF).....	118
TABLA 7: FACTOR DE AMBIENTE (EF).....	119
TABLA 8: ESTIMACIONES DE ESFUERZO POR ACTIVIDADES.....	121
TABLA 9: RESUMEN DE LOS RESULTADOS DE LA ESTIMACIÓN.....	122
TABLA 10: DESCRIPCIÓN DEL CUN “REALIZAR TESIS”.....	135
TABLA 11: DESCRIPCIÓN DEL CUN “ASIGNAR PERFIL DE TESIS”.....	136
TABLA 12: DESCRIPCIÓN DEL CUN “REALIZAR CORTE DE TESIS”.....	138
TABLA 13: DESCRIPCIÓN DEL CUS “AUTENTICAR USUARIO”.....	144
TABLA 14: DESCRIPCIÓN DEL CUS “GESTIONAR DOCUMENTO DE TESIS”.....	147
TABLA 15: DESCRIPCIÓN DEL CUS “GESTIONAR PERFIL DE TESIS”.....	150
TABLA 16: DESCRIPCIÓN DEL CUS “GESTIONAR TRIBUNAL DE TESIS”.....	154
TABLA 17: DESCRIPCIÓN DEL CUS “EVALUAR PERFIL DE TESIS”.....	156
TABLA 18: DESCRIPCIÓN DEL CUS “GESTIONAR PROYECTOS”.....	160
TABLA 19: DESCRIPCIÓN DEL CUS “GESTIONAR PROPUESTAS DE TESIS”.....	163
TABLA 20: DESCRIPCIÓN DEL CUS “GESTIONAR NOTAS DE LOS CORTES DE TESIS”.....	165
TABLA 21: DESCRIPCIÓN DEL CUS “GESTIONAR CORTES DE TESIS”.....	169
TABLA 22: DESCRIPCIÓN DEL CUS “GESTIONAR USUARIOS”.....	173
TABLA 23: DESCRIPCIÓN DEL CUS “REALIZAR COMENTARIOS”.....	174
TABLA 24: DESCRIPCIÓN DEL CUS “REALIZAR REPORTES”.....	174
TABLA 25: DESCRIPCIÓN DEL CUS “CONSULTAR COMENTARIOS”.....	176
TABLA 26: DESCRIPCIÓN DEL CUS “NOTIFICAR INFORMACIÓN DEL PERFIL DE TESIS”.....	177
TABLA 27: DESCRIPCIÓN DEL CUS “REVISAR DOCUMENTO DE TESIS”.....	178
TABLA 28: DESCRIPCIÓN DEL CUS “ELIMINAR DOCUMENTO DE TESIS”.....	179
TABLA 29: DIAGRAMA DE SECUENCIA DEL CUS “AUTENTICAR USUARIO”.....	180

TABLA 30: DIAGRAMA DE SECUENCIA DEL CUS “EVALUAR PERFIL”. SECCIÓN “ACEPTAR PERFIL”	181
TABLA 31: DIAGRAMA DE SECUENCIA DEL CUS “EVALUAR PERFIL”. SECCIÓN “RECHAZAR PERFIL”	182
TABLA 32: DIAGRAMA DE SECUENCIA DEL CUS “GESTIONAR CORTE DE TESIS”. SECCIÓN “ELIMINAR CORTE”	183
TABLA 33: DIAGRAMA DE SECUENCIA DEL CUS “GESTIONAR DOCUMENTO DE TESIS”. SECCIÓN “CARGAR DOCUMENTO”	184
TABLA 34: DIAGRAMA DE SECUENCIA DEL CUS “GESTIONAR PERFIL DE TESIS”. SECCIÓN “CREAR PERFIL (PROPUESTA)”	185
TABLA 35: DIAGRAMA DE SECUENCIA DEL CUS “GESTIONAR PERFIL DE TESIS”. SECCIÓN “MOSTRAR PERFIL”	186
TABLA 36: DIAGRAMA DE SECUENCIA DEL CUS “REALIZAR COMENTARIO”	187

Introducción

El desarrollo de las Tecnologías de la Información y las Comunicaciones (TIC) es incuestionable y forma parte de la cultura tecnológica que nos rodea y con la que debemos convivir. Amplía nuestras capacidades físicas y mentales, así como las posibilidades de desarrollo social. Las TIC tienen, día a día, una mayor presencia en todos los aspectos de la vida laboral y personal, ofreciendo un nuevo espacio de innovación en ámbitos como la industria, los servicios, la salud, la administración, el comercio y la educación.

La evolución de las tecnologías de la información tiende a incorporar, dentro de sus estructuras informáticas, la gestión de la documentación que tradicionalmente se ha conservado en los archivos, en papel.

Un sistema de gestión es una estructura probada para la gestión y mejora continua de las políticas, los procedimientos y procesos de una organización. [1]

En la actualidad, coexisten en el mundo los más diversos sistemas de gestión: desde el simple registro manual de la correspondencia, hasta los más sofisticados sistemas informáticos que manejan no sólo la documentación administrativa propiamente dicha, venga ella en papel o en formato electrónico, sino que además controlan los flujos de trabajo del proceso de tramitación de expedientes, capturan información desde bases de datos, contabilidad y otros, enlazan con el contenido de archivos, bibliotecas, centros de documentación y permiten realizar búsquedas sofisticadas y recuperar información.

En el ámbito de la educación el flujo de información que se debe controlar es voluminoso, lo que hace necesaria la existencia de los sistemas de gestión, sobre todo en la Educación Superior, etapa en la que se define la formación de profesionales, teniendo como aval de la misma, en gran parte de las carreras, un trabajo de diploma o tesis, el cual es imprescindible para lograr dicho objetivo, por lo que debe ser supervisado y evaluado periódicamente.

La palabra “tesis” proviene del latín “thesis” y significa “conclusión” que se mantiene por “razonamiento” [2]. Esta acepción se puede aplicar a una infinidad de propuestas, aunque comúnmente el significado en la Universidad, es que las tesis son la disertación escrita presentada para obtener el título de licenciatura, ingeniería, de maestría o doctorado.

La tesis es el documento en el que se exponen los resultados científicos alcanzados por el aspirante en su trabajo de investigación. Se presentan de forma sistematizada, lógica y objetiva esos resultados en correspondencia con el proyecto presentado, discutido y aprobado para la búsqueda de soluciones al problema planteado con respuestas científicas contextualizadas a partir de la utilización del método científico.[3]

En nuestro país, al igual que en el resto del mundo, es una meta el hecho de alcanzar un alto nivel de profesionalidad en los estudiantes que se encuentran en formación. Para esto, como se mencionaba anteriormente, es el nivel de enseñanza superior, el encargado de validar la categoría o especialidad alcanzada por cada uno de ellos, basándose en la discusión final de una tesis o trabajo de diploma. En dicho proceso es necesario un arduo trabajo en las diferentes instituciones para lograr una asignación de temas acorde con las características de los estudiantes y un control sistemático del trabajo que realicen para desarrollar estos temas.

Este proceso de asignación y control, se realiza hasta el momento de forma manual y requiere una inversión de tiempo y recursos que mayormente influye de forma negativa en la calidad y rapidez en la culminación de dichos trabajos.

La Universidad de Ciencias Informáticas (UCI), como centro de altos estudios, no está exento de dichas afectaciones, además de que por las características propias de la misma, se enfrenta a inconvenientes tales como el hecho de tener un número considerable de profesores externos, los cuales en cierto momento asumen la tutoría de tesis y no pueden realizar la supervisión de las mismas de la mejor manera, por no radicar en el centro. Los directivos de las facultades, encargados de realizar la asignación de los perfiles de tesis y evaluación de los mismos se enfrentan a situaciones como:

Se asignan perfiles de tesis que luego, por distintas razones, cambian o se eliminan y los encargados del proceso no están conscientes de dichos cambios.

Se realizan propuestas de perfiles por parte de los estudiantes, que mientras no se controlen pueden ser evaluados, sin embargo más adelante pudiera comprobarse que no tienen alcance como trabajo de diploma. Así en un momento dado aparecen estudiantes sin temas de tesis ya en un período avanzado del curso.

La revisión y corrección de los trabajos, por parte de los tutores, tiende a demorar el avance del mismo, ya que depende del tiempo disponible de las partes implicadas (tutor y tesista), para reunirse y realizar los señalamientos pertinentes.

No se cuenta con un control preciso sobre los profesores vinculados a las tesis, ni respecto a los que han sido ubicados en comisiones de tesis, de modo que no es posible determinar quiénes de ellos se encuentran disponibles para desempeñar una u otra tarea.

La información relacionada con las evaluaciones de las tesis en cada corte realizado no se maneja de manera centralizada, o sea, cada comisión de tesis, controla estos datos sin que los miembros del comité de tesis estén conscientes del estado (bien, regular, mal) en que se encuentran los estudiantes.

La Facultad 2 también se ha visto afectada por los problemas mencionados, de modo que se plantea un **problema científico** determinado por la falta de controles detallados respecto al estado evolutivo de los trabajos de tesis y del personal disponible para asumir los roles requeridos durante la etapa de desarrollo del proceso de tesis en la Facultad 2. Por tanto, el **objeto de estudio** de la investigación son los procesos de gestión de tesis, definiendo como **campo de acción** de la misma el proceso de gestión de las tesis en la Facultad 2. No se tiene un control preciso del personal disponible para asumir tutorías, ni del estado evolutivo de los trabajos de tesis

De esta manera, la **idea a defender** con la presente investigación, es que la realización de un sistema de gestión de tesis, ajustada a las necesidades, estructura y estrategia de trabajo de la Facultad 2, mejoraría el proceso de asignación, revisión y control de las mismas.

El **objetivo general** de este trabajo es desarrollar una aplicación web que permita mejorar el manejo de las tesis en la Facultad 2.

Para desarrollar el objetivo general se plantean los siguientes **objetivos específicos**:

1. Realizar un estudio del estado del arte sobre las principales tendencias de la gestión de tesis, así como sus características en la Facultad 2.
2. Escoger una metodología y seleccionar las herramientas adecuadas para desarrollar el sistema de gestión propuesto.

3. Analizar y diseñar la propuesta de solución para gestionar las tesis de la Facultad 2.
4. Desarrollar el sistema de gestión haciendo uso de las técnicas y herramientas seleccionadas anteriormente.

Para darle solución al problema científico planteado y lograr el cumplimiento de los objetivos se proponen las siguientes tareas:

1. Realizar un estudio del estado del arte teniendo en cuenta los trabajos anteriores relacionados con el tema.
2. Investigar sobre el proceso de gestión de tesis en otras facultades.
3. Establecer una comparación entre las diferentes formas de realizar este proceso en la UCI, para proponer una mejora sustancial al desarrollo del mismo.
4. Realizar un estudio sobre las metodologías posibles a utilizar.
5. Realizar un estudio de las principales tendencias y tecnologías actuales a utilizar.
6. Analizar, diseñar e implementar el sistema propuesto.
7. Realizar un estudio de la factibilidad técnica.

El contenido del presente documento se estructura en cinco capítulos, donde se incluye todo lo relacionado con el trabajo desarrollado.

Capítulo 1 “Fundamentación Teórica”. Aborda el estudio del estado del arte del tema relacionado con el trabajo que se desarrolla. Se destacan además las tendencias y tecnologías actuales sobre las cuales se apoya la propuesta del sistema informático, así como las diferentes metodologías de desarrollo de software y herramientas analizadas hasta mostrar las escogidas.

Capítulo 2 “Características del Sistema”. Se describe y modela el Negocio, obteniendo como resultado los artefactos correspondientes a este flujo de trabajo, como son el Diagrama de Casos de Uso del Negocio, los actores y trabajadores del mismo. Se especifican los requerimientos funcionales y no

funcionales, se realiza el Diagrama de Casos de Usos del Sistema y se muestra una propuesta del prototipo de interfaz para el sistema a desarrollar.

Capítulo 3 “Análisis y Diseño del Sistema”. Presenta el Modelo de Análisis, en el cual se desarrolla el Diagrama de Clases del Análisis; el Modelo de Diseño, que incluye el Diagrama de Clases del Diseño Web y los diagramas de interacción por casos de uso (CU), específicamente los diagramas de secuencia; además se realiza el Modelo de Datos con el Diagrama Entidad Relación.

Capítulo 4 “Implementación”. Define cómo se organizan las clases y objetos en componentes, cuáles nodos se utilizarán y la ubicación en ellos de los componentes, estructurando así la aplicación.

Capítulo 5 “Estudio de la Factibilidad”. En este capítulo se realiza una estimación del esfuerzo, el tiempo y el costo para determinar la factibilidad del sistema.

Capítulo 1. Fundamentación Teórica

1.1 Introducción

En este capítulo se realiza una investigación relacionada con los sistemas de gestión, incluyendo el estado del arte a nivel mundial, en Cuba y en la UCI y las tendencias actuales. También se exponen las principales características de las tecnologías y herramientas seleccionadas para la solución del problema planteado, así como la justificación de la selección de las mismas.

1.2 Estado del Arte

1.2.1 Sistemas de Gestión

El término Sistema de Gestión abarca un amplio campo, dado que la gestión y planificación son una necesidad desde los inicios de las organizaciones, empresas e instituciones de diversas índoles. Un sistema de este tipo puede ser concebido para múltiples propósitos según la esfera en que se desee implantar, o sea, puede gestionarse información referente a, por solo citar algunos, empresas, escuelas, institutos de investigación y datos de logística. Por tal razón los mismos pueden clasificarse según su propósito en (solo por enumerar unas pocas clasificaciones):

- ✓ Sistema de Gestión de Contenido, concebido desde un punto de vista para suplir necesidades más genéricas. Es una aplicación que sirve para simplificar el proceso de crear, publicar y gestionar los contenidos de un sitio web, como es el caso de Drupal, el cual permite publicar artículos, imágenes, u otros archivos y servicios añadidos como foros, encuestas, votaciones, blogs y administración de usuarios y permisos.
- ✓ Sistema de Gestión de Calidad, concebido con el objetivo de orientar las actividades de una empresa para obtener y mantener el nivel de calidad del producto o el servicio, de acuerdo con las necesidades del cliente, proporcionando actividades coordinadas para dirigir y controlar una organización en lo relativo a la calidad. En este caso se puede citar el ejemplo de Q-bo.org, que es un sistema avalado y

recomendado por las entidades de certificación a nivel internacional, que se dedica a mejorar la eficacia de los sistemas de Calidad ISO 9001¹.

- ✓ Sistema de Gestión Comercial, destinado a esbozar el mapa de la geografía comercial que vive una empresa, y que le permite anticipar caminos y obstáculos, dándole la información indispensable para la toma de decisiones, como lo es ERP (Enterprise Resource Planning- Planificación de Recursos Empresariales), que además de la gestión tradicional, brinda la facilidad de presupuestar, sacar estadísticas, generar proyecciones en base a históricos, lo que permite optimizar el performance comercial. Algunos ejemplos de ERP lo constituyen SAP (Systems, Applications and Products in Data Processing), el cual tiene su sede en Walldorf, Alemania y, Microsoft AXAPTA. Este último es el software ERP desarrollado por Microsoft. Axapta es una solución de gestión integrada que cubre todas las áreas de negocio de la empresa. Este software ERP es personalizable, escalable y global, y facilita la comunicación con el entorno comercial. [4] Varias empresas en Cuba ya tienen sistemas nacionales de gestión comercial y además se ha comenzado un proyecto para la creación de un ERP cubano, que está siendo desarrollado en la UCI.
- ✓ Sistema de Gestión de Producción, concebido para centrar, organizar, sistematizar los procesos y cualquier actividad empresarial que implique manejar una cantidad significativa de información relativa a la clientela, empresas proveedoras, ventas, entre otros. MRP (Manufacturing Requirement Planning -Planificación de Requerimientos Materiales), es un ejemplo de este tipo de sistemas, dedicados a integrar las actividades de producción y compras. Es un sistema de planificación de la producción y de gestión de stocks (o inventarios) que responde a las preguntas: ¿qué? ¿cuánto? y ¿cuándo?, se debe fabricar y/o aprovisionar. El objetivo del MRP es brindar un enfoque más efectivo, sensible y disciplinado para determinar los requerimientos de materiales de una empresa: DX Software constituye un ejemplo de los sistemas MRP. En el sitio oficial del mismo, se expresa que este software ofrece una serie de soluciones en materia de software de gestión de producción, basado en módulos referidos a

¹ La Norma ISO 9001 ha sido elaborada por el Comité Técnico ISO/TC176 de ISO (Organización Internacional para la Estandarización) y especifica los requisitos para un sistema de gestión de la calidad que pueden utilizarse para una aplicación interna, por las organizaciones, para certificación o con fines contractuales.

stocks y a la producción, como costos estándares, precios de venta, explosión e implosión de productos, simulación de compras, entre otros.

- ✓ Sistema de Gestión Documental (SGD), concebido para el manejo de los documentos en cualquier formato y soporte, producidos o recibidos por una organización. En este caso se encuentra Inmagic, especializado en la gestión de información textual no estructurada y/o en la gestión de documentos científicos y técnicos, con prestaciones suficientes para automatizar las operaciones más importantes de la cadena documental.

Los sistemas de gestión, ya sean de uno u otro tipo, contribuyen a gestionar, por ejemplo, los riesgos sociales, medioambientales y financieros, mejorar la efectividad operativa, reducir costos y lograr mejoras continuas. Es por eso que en nuestro país, también se han implementado sistemas de este tipo, como es el caso de VERSAT-Sarasola, un paquete integrado para la gestión económica financiera que permite enviar información eficaz, de forma inmediata, desde lugares apartados, a la vez que ofrece mayor organización, control y disciplina en cada gestión. Fue creado por el villaclareño Miguel Cabrera González y se aplicó por primera vez en el central azucarero George Washington, en 2001. Fue el primer sistema cubano que logró certificarse en el país y está estructurado con los elementos requeridos por la contabilidad de este, con lo cual se ha ganado la aceptación en el mercado y ha logrado diseminarse por un gran número de entidades nacionales.

SLD012 es otro software de gestión cubano desarrollado para procesar los datos clínicos del Proyecto Mapeo Cerebral Humano Cubano (MCHC), coordinado por el Centro de Neurociencias de Cuba y dirigido a crear un ATLAS de la estructura y funcionamiento cerebral, aportando así, conocimientos sobre los sustratos neurales tanto de la función normal como sus alteraciones en las enfermedades neuropsiquiátricas. SLD012 es una aplicación Web que permite gestionar con mayor agilidad y correctamente toda la información obtenida en los exámenes clínicos, haciendo uso de instrumentos avalados internacionalmente que dan validez al criterio del diagnóstico arrojado por el examen.

También se cuenta con un sistema de conservación, gestión y socialización documental para los archivos en Cuba, llamado Papiro. Este producto informático es de uso libre, emplea herramientas igualmente libres y permite conservar documentación de valor histórico, al evitar su manipulación, pues al digitalizarse el documento, éste se consulta en formato electrónico. Fue concebido para desarrollar y establecer, en el

ámbito de los Archivos de Cuba, los procedimientos y estándares para el procesamiento digital del patrimonio de base documental, con el objetivo de contribuir a la preservación y divulgación de dicho acervo, facilitar el proceso investigativo y conocimiento de la historia y cultura cubanas a partir de una amplia socialización documental y ofrecer una herramienta que contribuya a una mayor eficacia de la gestión administrativa.

Sin embargo, a pesar de haberse desarrollado los sistemas mencionados y otros muchos, aún en el área de la educación, sobretodo en la educación superior, no se cuenta en Cuba con las aplicaciones informáticas necesarias para mejorar los procesos de gestión de los eventos que se desarrollan a este nivel, como es el caso de las tesis.

1.2.2 Sistemas de Gestión de Tesis

Como hasta el momento se ha demostrado, la gestión ha sido columna vertebral de las instituciones en el transcurso del tiempo. En un marco aún más específico los sistemas de gestión se convierten en pilar imprescindible, a la vez que por su parte las aplicaciones creadas para este fin, contribuyen a satisfacer las exigencias que imponen los avances en el aspecto empresarial, cultural, científico técnico, educacional, y otras esferas.

Para ello en un sector particular como el educacional, se precisa un nivel de optimización en el proceso de gestión, más aún en el caso de la Educación Superior, que como se había mencionado anteriormente, es la que se responsabiliza en la formación especializada de profesionales y que en algunos casos valida la terminación de estudios por parte de los cursantes mediante la confección y discusión final de un trabajo de diploma (tesis).

Evidentemente no en todos los centros se gestiona el mismo de tipo de recursos informacionales, tal es el caso de que en el ambiente universitario lo más importante es gestionar tanto la información referente a las investigaciones llevadas a cabo por muy diversas que sean, como la documentación de los procesos que se efectúen en el mismo, la relacionada con la gestión académica, la matrícula, además de la referente a los trabajos de diploma; para este fin es posible especializar un sistema de gestión para tesis.

Independientemente de la estrategia que lleve cada centro de Educación Superior para controlar el desarrollo de las tesis, teniendo en cuenta, por ejemplo su especialidad o estructura organizacional, en la

actualidad el auge de las TICs exigen realizar dicho control de forma automatizada y, para ello nada más recomendable que los sistemas especializados en este tipo de funciones, díganse Sistemas de Gestión de Tesis propiamente dicho.

Se pueden citar algunos ejemplos de centros que hacen uso de software de este tipo para realizar sus funciones. Tal es el caso de algunas universidades europeas como:

Universidad de Castilla-La Mancha, en esta universidad se implementó la aplicación informática **TESIS**, que fue desarrollada por los Servicios Informáticos de la Universidad de Castilla-La Mancha en el año 1998, a propuesta del Archivo General Universitario, en colaboración con la Secretaría General y el Vicerrectorado de Ordenación Académica. La aplicación permite realizar, compartida y automatizadamente, la gestión administrativa de las tesis doctorales presentadas en la institución. Mediante TESIS se controla el trámite y la localización física de la tesis doctoral a lo largo de todo el procedimiento para su lectura, desde el trámite de presentación al de archivo, incluyendo además otras posibilidades como es el análisis del contenido o la información relativa a la posterior publicación de estos trabajos de investigación.

Las funciones de la aplicación informática TESIS se pueden resumir, atendiendo a la finalidad que cumplen en lo que se ha convenido denominar unidades administrativas, del siguiente modo:

Secretaría General:

- ✓ Registro de tesis doctorales admitidas a trámite para su lectura.
- ✓ Control del trámite de exposición pública y lectura de tesis.
- ✓ Control de la remisión al Archivo General de la Universidad para su archivo.

Vicerrectorado de Ordenación Académica:

- ✓ Control de la tramitación y contenido de las tesis doctorales.

Archivo General Universitario:

- ✓ Control de la transferencia y recepción de las tesis doctorales.

- ✓ Control del almacenamiento de las tesis doctorales.
- ✓ Control del almacenamiento de los expedientes de lectura de tesis doctorales.
- ✓ Control de la información contenida en las tesis doctorales.
- ✓ Control de la publicación de las tesis doctorales.
- ✓ Control de la consulta de las tesis doctorales. [5]

En la Facultad de Educación de la Universidad de Salamanca, concretamente desde el laboratorio de diseños multimedia y teleeducación, se desarrolló también, como parte del programa global de estudios, diseño y producción de espacios virtuales, el sistema “Gestor de Tesis”. (Ver figura 1.1).



Figura 1: Pantalla principal de la aplicación Gestor de Tesis

Esta herramienta permite al tutor el seguimiento de las tesis que dirige, incorporando a la base de datos los trabajos de los cuales es director y los doctorandos que las tienen asignadas. Para cada tesis se recogen los documentos relacionados, la bibliografía recomendada, enlaces a sitios en Internet y mensajes de correo electrónico que se intercambien con el doctorando, el cual tiene igualmente acceso a estos recursos.

Las funcionalidades más importantes son:

- ✓ Gestión de las tesis en realización
- ✓ Seguimiento del trabajo de los doctorandos
- ✓ Intercambio de la documentación disponible en formato electrónico
- ✓ Base de datos de referencias bibliográficas
- ✓ Enlaces a otros programas de fuentes bibliográficas de la Universidad de Salamanca
- ✓ Registro de los mensajes de correo electrónico intercambiados[6]

La particularidad de los sistemas de Gestión de Tesis aun no encuentra un amplio uso en la Educación Superior en Cuba. Se mantienen métodos y estrategias tradicionales en este aspecto, independientemente de las características de cada centro en esencia, como por ejemplo el perfil de las carreras que se imparten (dígase en este caso que una Universidad donde solo se realizan Pruebas Estatales para culminar los estudios, no se precisa este tipo de sistema) y la infraestructura tecnológica.

Pero se han dado pasos en aras de lograrlo, es por eso que teniendo en cuenta los aspectos anteriormente mencionados, en la Universidad de las Ciencias Informáticas (UCI), dadas las condiciones de la misma, se manifiesta una corriente creciente al uso de los sistemas de gestión tanto para los procesos administrativos como docentes y de producción. Con la imperiosa necesidad de informatización, se impone de esta manera la tarea de agilizar el proceso de revisión y control de las tesis, como una labor importante en el aspecto docente.

Por citar un ejemplo del uso de estos sistemas en la UCI podemos mencionar una aplicación desarrollada en la Facultad 8, la cual en principio fue concebida haciendo uso de tecnologías propietarias, además de que por la forma en que el mismo fue diseñado no se podía publicar la información que almacenaba para que estuviese al alcance de profesores y estudiantes. Luego en un proceso de mejora se trasladó todo el contenido de este sistema para una aplicación Web dinámica incorporándole nuevas funcionalidades, Aplicación Web para el control del Proceso de Tesis de Grado en la Facultad 8. Este sistema se realizó con la idea de resolver las limitantes anteriores y de poner a disposición de la Facultad 8 una herramienta que permitiera la automatización de toda la información generada en el proceso de tesis de grado. El mismo permite realizar propuestas de tesis, gestionar los tribunales, cortes y perfiles, mantener un seguimiento sobre las evaluaciones y realizar señalamientos por parte del comité de tesis.

También se cuenta en la Facultad 6 con el Sistema para la Gestión de las Tesis, el cual se enmarca en la fase informativa más que en la de gestión de las tesis en esencia. Este sistema cuenta con secciones de planificación, por ejemplo, para las reuniones de los comités de tesis; así como informativas, sobre qué actividades desempeña un tutor, cuáles desempeña un oponente, la estructura de un documento de tesis según el tipo de la misma. Por otra parte permite crear encuestas, paneles de discusión y pone a disposición de los usuarios documentos compartidos como ejemplos de tesis, manuales, metodologías y técnicas. Esta aplicación tiene como inconveniente el hecho de que se encuentra implementada sobre la plataforma .NET.

Además se puede mencionar el sitio central Tesis.uci.cu el cual tiene el objetivo de informar y de asistir a los estudiantes de 5to año durante el desarrollo de sus trabajos de diploma.

Estas aplicaciones han hecho aportes en cuanto a los pasos hacia la implementación de los sistemas de gestión de tesis en la UCI de acuerdo al marco en que han sido utilizados, sin embargo están muy lejos aún de las expectativas que se han trazado en este campo para la Facultad 2. Por tal motivo se ha decidido implementar el sistema Tesis.Fac2, que comparte ciertas semejanzas con los ya existentes, pero sobretodo, ajustado a nuestros intereses, estructura organizacional, estrategia de trabajo, perfil de estudio, entre otros aspectos a tener en cuenta. Lo primero que se pretende lograr con esta aplicación es proporcionar un sistema totalmente libre de herramientas propietarias que ayude a mejorar el proceso de gestión de tesis, que brinde facilidades como, el hecho de mostrar un número de reportes considerables que le permitan al personal involucrado en el manejo de las tesis, llevar un constante y detallado control de las mismas, de modo que el flujo de información pueda hacerse de forma no presencial, tener constancia de las evaluaciones obtenidas en los diferentes cortes, así como en determinado momento hacer una propuesta de nota final para cada equipo de tesis. Además de que estará disponible tanto para el personal de dirección: jefes de tribunales de discusión, vicedecano de producción, decano docente; como para los tutores, y tesistas. O sea, que en principio no es una aplicación restringida solo al alcance de la dirección de la facultad o para agilizar el trabajo de la misma, sino que es de alcance público. Los tutores podrán acceder y realizar las revisiones pendientes a las tesis, haciendo los señalamientos pertinentes sin necesidad de que tengan concertar encuentros con los tesistas para debatir las modificaciones recomendadas.

Además se brindará la posibilidad de que los tutores elaboren nuevos perfiles de tesis a partir de propuestas realizadas por los estudiantes, los cuales posteriormente pueden ser enviados al personal correspondiente para que confirmen la aprobación o no del tema.

Dada la necesidad de interacción entre la aplicación y los usuarios por las razones anteriormente expuestas, se decidió desarrollar una aplicación web dinámica que cumpla con las características de una Web 2.0².

1.3 Justificación de la Propuesta

1.3.1 Aplicaciones Web

Una aplicación Web es un sitio Web donde la interacción a través del sitio y la incorporación de datos por parte de los usuarios afectan el estado de la lógica del negocio del mismo; es decir, el sitio Web procesa los datos introducidos por los usuarios y les muestra a estos un resultado en dependencia de la información introducida. Si no existe un procesamiento diferente en el servidor en dependencia de la petición del usuario, el sitio Web no puede ser considerado como aplicación Web. [7]

Las aplicaciones Web son hoy en día uno de los sistemas más comunes que podemos encontrar en la red a través de Internet. Ejemplo de estas aplicaciones Web son los sistemas como GMAIL y HOTMAIL que brindan la posibilidad de establecer comunicación a través de correo electrónico o mensajería instantánea (o ambos) con personas en todo el mundo, y existen otros como Amazon, que permiten comprar productos en internet o Google para buscar información.

Las aplicaciones web constan de tres niveles fundamentales:

- ✓ Interfaz de usuario.
- ✓ Lógica de negocio.
- ✓ Datos.

² La Web 2.0 es la representación de la evolución de las aplicaciones tradicionales hacia aplicaciones web enfocadas al usuario final.

El nivel de interfaz de usuario está compuesto por documentos escritos en lenguaje HTML con los datos que se han obtenido de un servidor web³ y se muestran a los usuarios mediante un cliente web⁴ (generalmente un navegador web).

El nivel de lógica de negocio está constituido por módulos que implementan la lógica de negocio. Esta se divide en dos: lógica de la aplicación y lógica del dominio, la cual se ejecuta en un servidor de aplicaciones. En la actualidad los servidores Web funcionan a la vez como servidor de aplicación.

El nivel de datos está integrado por la información, que generalmente está manejado por un sistema de gestión de base de datos, el cual es accedido mediante el servidor de aplicaciones.

Las aplicaciones Web constan de un cliente (navegador), un servidor (servidor Web) y una red que posibilita la comunicación entre el cliente y el servidor, de manera parecida al modelo cliente-servidor, el cual es perfectamente aplicable a la solución que se pretende lograr.

1.3.2 Modelo Cliente – Servidor

La utilización de las diferentes aplicaciones o servicios de Internet se lleva a cabo respondiendo al llamado modelo cliente-servidor, también conocido como arquitectura cliente-servidor, que es una forma de especializar terminales y programas de forma que las actividades y tareas que cada uno de ellos realice se efectúe con la mayor eficiencia posible. Dicha arquitectura es cualquier combinación de sistemas que puedan colaborar entre sí para dar a los usuarios toda la información que ellos necesiten, sin que tengan que saber dónde está ubicada.

El escenario de un modelo cliente-servidor ocurre cuando el programa cliente se encarga de ponerse en contacto con el ordenador remoto para solicitar el servicio deseado. El ordenador remoto por su parte responderá a lo solicitado por el programa cliente mediante otro programa, denominado programa servidor (**Ver figura 1.2**). Los elementos principales del modelo cliente-servidor son justamente los elementos llamados cliente, servidor y red de comunicación.

³ Servidor web, sirve contenido estático a un navegador, carga un archivo y lo sirve a través de la red al navegador de un usuario mediante el protocolo HTTP.

⁴ Cliente web, es cualquier aplicación que sirve para utilizar la web.

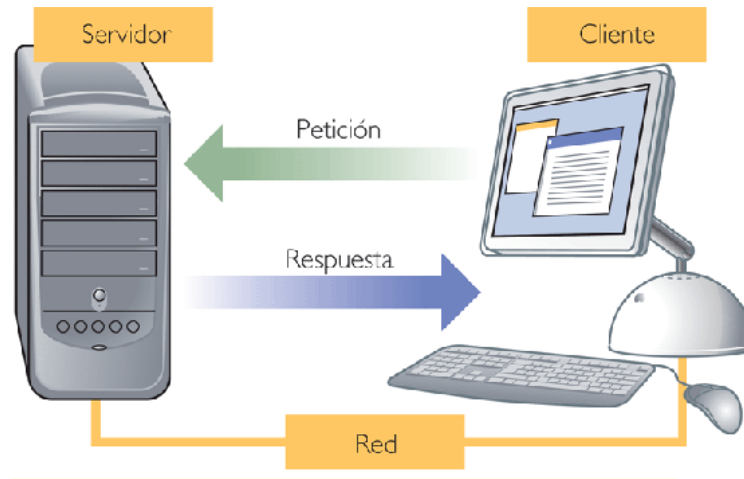


Figura 2: Modelo cliente - servidor

El diálogo cliente-servidor es casi siempre bidireccional. Por un lado, el cliente envía información al servidor (el tipo de servicio solicitado más los parámetros); por otro, el servidor devuelve información al cliente (los resultados del servicio, códigos de error en caso de producirse) [8]

Las características principales que identifican al modelo son:

1. Pueden actuar como una sola entidad y también como entidades separadas, realizando actividades y tareas separadas.
2. Las funciones del cliente y el servidor pueden estar en plataformas separadas, o en la misma plataforma.
3. Un servidor da servicio a múltiples clientes de forma concurrente.
4. Cada plataforma puede ser escalable e independiente. Los cambios realizados en las plataformas de los clientes o de los servidores, ya sean por actualización o por reemplazo tecnológico, se realizan de una manera transparente para el usuario final.
5. La interrelación entre el hardware y el software están basados en una infraestructura poderosa, de tal forma que el acceso a los recursos de la red no muestra la complejidad de los diferentes tipos de formato de datos y de los protocolos.

6. Un sistema de servidores realiza múltiples funciones y tareas, al mismo tiempo que presenta una imagen de un solo sistema a las estaciones clientes. [8]

Los términos cliente y servidor se utilizan tanto para referirse a los programas que cumplen estas funciones, como a los ordenadores donde son ejecutados esos programas.

1.3.2.1 Tecnologías del lado del cliente

Los programas clientes utilizados para acceder a los servicios realizan dos funciones distintas. Por una parte, se encargan de gestionar la comunicación con el ordenador servidor, de solicitar un servicio concreto y de recibir los datos enviados por éste; y por otra, es la herramienta que presenta al usuario los datos en pantalla y que le ofrece los comandos necesarios para utilizar las prestaciones que ofrece el servidor. A continuación se muestran las tecnologías utilizadas para desarrollar la aplicación que se ejecutan como programas clientes:

➤ HTML

HTML significa HyperText Markup Language. Es el lenguaje en que se escriben los millones de documentos que hoy existen en el World Wide Web. Cuando accedemos a uno de estos documentos, el cliente (Netscape, IE, Mosaic, Lynx, Mozilla) los interpreta y los muestra. [9]

El HTML no es más que una aplicación del Standard Generalized Markup Language (SGML), un sistema para definir tipos de documentos estructurados y lenguajes de marcas⁵ para representar esos mismos documentos. El término HTML se suele referir a ambas cosas, tanto al tipo de documento como al lenguaje de marcas.

Además está diseñado para estructurar textos y presentarlos en forma de hipertextos⁶, que es el formato estándar de las páginas web. HTML es un lenguaje interpretado, por lo que necesita de un programa que

⁵ Lenguaje de marcas, es una forma de codificar un documento que, junto con el texto, incorpora etiquetas o marcas que contienen información adicional acerca de la estructura del texto o su presentación.

⁶ Hipertexto, es el nombre que recibe el texto que en la pantalla de una computadora conduce a su usuario a otro texto relacionado.

interprete las etiquetas que están dentro del cuerpo del documento en formato HTML y los muestre en el navegador.

La principal ventaja que tiene HTML es la gran variedad de navegadores y exploradores que lo soportan. Debido a esto, se ha convertido en el formato más usado para la transmisión de documentos electrónicos a través de Internet. Otra ventaja que aporta es su compatibilidad con los navegadores más antiguos, ya que estos navegadores aún pueden interpretar documentos electrónicos en HTML. Otro beneficio que aporta es que disminuye el costo de aprendizaje de las personas que empiezan a aprender a leer y a escribir HTML, siendo un lenguaje muy fácil de utilizar y comprender.

➤ **CSS**

CSS significa Hojas de Estilo en Cascada (Cascading Style Sheets). Es un mecanismo simple que describe cómo se va a mostrar un documento en la pantalla, o cómo se va a imprimir, o incluso cómo va a ser pronunciada la información presente en ese documento a través de un dispositivo de lectura. Esta forma de descripción de estilos ofrece a los desarrolladores el control total sobre estilo y formato de sus documentos. [10]

CSS se utiliza para dar estilo a documentos HTML y XML, separando el contenido de la presentación. Los estilos definen la forma de mostrar los elementos HTML y XML. CSS permite a los desarrolladores Web controlar el estilo y el formato de múltiples páginas Web al mismo tiempo. Cualquier cambio en el estilo marcado para un elemento en la CSS afectará a todas las páginas vinculadas a esa CSS en las que aparezca ese elemento.

El navegador, para poder interpretar las CSS, debe estar embebido dentro de cualquiera de los formatos existentes para confeccionar documentos electrónicos para la web.

Unos de los principales beneficios que aporta el uso de las Hojas de Estilos en Cascada es el control de diseño, ya que permite separar el contenido del mismo, de modo que para cambiar un aspecto del diseño de un sitio web, solo es preciso modificar la hoja de estilo, con lo que se logra un ahorro de tiempo y una mayor uniformidad en el diseño.

Otra ventaja que aporta la utilización de CSS es que permite que las páginas web se presenten en buen estado en los navegadores, permitiendo que se elaboren documentos electrónicos bien definidos. Permiten a los diseñadores reducir los costos de mantenimiento de las páginas web al reducir la complejidad de los documentos.

➤ **JavaScript**

JavaScript es un lenguaje de scripts⁷ desarrollado por Netscape para incrementar las funcionalidades del lenguaje HTML. Este es un lenguaje interpretado, es decir, no requiere compilación. El navegador del usuario se encarga de interpretar las sentencias JavaScript contenidas en una página HTML y ejecutarlas adecuadamente.

Las dos principales características de JavaScript son, por un lado que es un lenguaje basado en objetos (es decir, el paradigma de programación es básicamente el de la programación dirigida a objetos, pero con menos restricciones), y por otro que es orientado a eventos. Esto implica que gran parte de la programación en JavaScript se centra en describir objetos y escribir funciones que respondan a movimientos del mouse, pulsación de teclas, apertura y cerrado de ventanas o carga de una página, entre otros eventos. [11]

Una de las principales ventajas del uso de JavaScript es que permite elaborar aplicaciones Web que simulen características de aplicaciones escritorios, ya que logra establecer una mejor interacción entre las páginas web y los usuarios. Otra ventaja que aporta JavaScript es que se puede utilizar en la elaboración de páginas web para ser vistas desde cualquier plataforma (Windows, Linux), debido a que es soportado por una gran variedad de navegadores y exploradores Web. JavaScript también le permite a los desarrolladores de interfaces de usuario tener mejor control de las páginas Web. Esto se logra gracias a que a través de JavaScript, los desarrolladores de interfaces de usuario pueden acceder a las diferentes partes y elementos de una página Web (botones, textos, imágenes, videos, etiquetas).

⁷ Un lenguaje de script es un lenguaje de programación que un programa de ordenador puede interpretar. En el caso a que se hace referencia el código se inserta dentro del documento HTML.

Lamentablemente el lenguaje JavaScript no resuelve todas las expectativas de los desarrolladores. En ocasiones los desarrolladores deben realizar diferentes implementaciones de código JavaScript para sus aplicaciones, debido a que no todos los navegadores interpretan de la misma manera el código. Unos de los inconvenientes que tiene este lenguaje es que tiene que estar activado en los navegadores para que su uso sea posible por estos.

➤ **AJAX**

El término AJAX es un acrónimo de Asynchronous JavaScript + XML (Extensible Markup Language), que se puede traducir como "JavaScript asíncrono + XML". Esta no es una tecnología en sí misma. En realidad, se trata de varias tecnologías independientes que se unen de formas nuevas y sorprendentes [12] que pueden lograr cosas realmente impresionantes como GoogleMaps, Gmail el Outlook Web Access o algunas otras aplicaciones muy conocidas. Algunas de estas tecnologías son:

- ✓ XHTML y las hojas de estilos en cascada (CSS) que se usan para el diseño que acompaña a la información.
- ✓ JavaScript que permite acceder al DOM (Document Object Model) para mostrar e interactuar dinámicamente con la información presentada.
- ✓ El objeto XMLHttpRequest, el cual es utilizado para intercambiar datos de manera asíncrona entre el navegador y el servidor.
- ✓ XML y JSON que son los formatos que comúnmente se utilizan para la transferencia de datos devueltos por el servidor. También las dos tecnologías pueden ser usadas como formato de transferencia desde el navegador Web y el servidor.

AJAX es una técnica de desarrollo Web para elaborar aplicaciones interactivas que se ejecutan en el cliente, es decir, en el navegador del usuario, y mantienen comunicación asíncrona con el servidor en segundo plano. De esta forma es posible realizar cambios sobre la misma página Web sin necesidad de recargarla, lo que significa aumentar la interactividad, velocidad y eficiencia de las aplicaciones basadas en la Web, mejorando el tiempo de respuesta. Además, aunque no es soportado por todos los

navegadores, es válida en cualquier plataforma, no es difícil de usar y es independiente del tipo de tecnología que se utilice en el servidor.

➤ **DOM**

Modelo de Objetos del Documento (DOM) es un conjunto de utilidades específicamente diseñadas para manipular documentos XML. Por extensión, DOM también se puede utilizar para manipular documentos XHTML y HTML. Técnicamente, DOM es una API de funciones que se pueden utilizar para manipular las páginas XHTML de forma rápida y eficiente.

Antes de poder utilizar sus funciones, DOM transforma internamente el archivo XML original en una estructura más fácil de manejar formada por una jerarquía de nodos. De esta forma, DOM transforma el código XML en una serie de nodos interconectados en forma de árbol.

Aunque en ocasiones DOM se asocia con la programación web y con JavaScript, la API de DOM es independiente de cualquier lenguaje de programación. De hecho, DOM está disponible en la mayoría de lenguajes de programación comúnmente empleados. La ventaja de emplear DOM es que permite a los programadores disponer de un control muy preciso sobre la estructura del documento HTML o XML que están manipulando. Las funciones que proporciona DOM permiten añadir, eliminar, modificar y reemplazar cualquier nodo de cualquier documento de forma sencilla. [13]

1.3.2.2 Tecnologías del lado del servidor

En la mayoría de los casos se opta por utilizar tecnologías del lado del servidor, por varios motivos, por ejemplo: al ejecutarse en el servidor las respuestas son, por lo general, páginas basadas en tecnologías estandarizadas, de modo que cualquier navegador puede interpretarlas, cosa que no pasa con las tecnologías cliente. Además de tener en cuenta que se contribuye a la seguridad, puesto que al ejecutarse el código fuente en el servidor, la programación es transparente al cliente, permitiendo ocultar así los detalles de implementación. [14]

A continuación se muestran, entre algunas de las tecnologías que se ejecutan en el equipo servidor, las utilizadas para desarrollar la aplicación:

➤ **Gestor de Base de Datos**

Existen varias formas de guardar información necesaria y de vital importancia, una de las más conocidas y seguras son las bases de datos.

Una Base de Datos (BD) es un conjunto de datos interrelacionados entre sí, almacenados con carácter más o menos permanente en la computadora, o sea, que puede considerarse una colección de datos variables en el tiempo. [15]

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, se denomina Sistema de Gestión de Bases de Datos (SGBD), cuyo objetivo fundamental consiste en suministrar al usuario las herramientas que le permitan manipular los datos en términos abstractos, de forma que no necesite conocer el modo de almacenamiento de los mismos en la computadora, ni el método de acceso empleado.

Los programas de aplicación operan sobre los datos almacenados en la base de datos, utilizando las facilidades que brindan los SGBD, los que, en la mayoría de los casos, poseen lenguajes especiales de manipulación de la información que facilitan el trabajo de los usuarios.

Entre los SGBD comúnmente utilizados en el mundo tenemos Oracle, MySQL, Microsoft SQL Server, PostgreSQL, Interbase, entre otros. Todos estos presentan un enfoque relacional con un buen basamento matemático centrado en el Álgebra Relacional.

Todos los sistemas mencionados anteriormente facilitan el trabajo con la base de datos y tienen características que los diferencian, por ejemplo:

- ✓ Oracle es básicamente una herramienta cliente/servidor para la gestión de bases de datos que requiere de una licencia privativa para poderlo utilizar. Se considera como uno de los sistemas de bases de datos más completos, destacando su estabilidad, escalabilidad, soporte de transacciones y soporte multiplataforma. Cuenta con mecanismos de seguridad, determinados por el acceso a los datos según los privilegios concedidos por el administrador, proporciona sofisticados procedimientos para hacer copias de seguridad y recuperar datos, brinda la posibilidad de asignar espacio en disco duro para almacenarlos y controlar dicho espacio.

Oracle es un producto vendido a nivel mundial, aunque la gran potencia que tiene y su elevado precio, hacen que sólo se vea en empresas muy grandes y multinacionales por norma general, en resumen, es necesario pagar para poder utilizarlo.

- ✓ Microsoft SQL Server es un sistema de gestión de bases de datos relacionales capaz de poner a disposición de muchos usuarios grandes cantidades de datos de manera simultánea. Este SGBD garantiza soporte de transacciones, escalabilidad, estabilidad y seguridad, soporta procedimientos almacenados, incluye también un potente entorno gráfico de administración, que permite el uso de comandos DDL(Data Definition Lenguaje) y DML(Data Manipulation Lenguaje) gráficamente, permite trabajar en modo cliente-servidor, donde la información y datos se alojan en el servidor y las terminales o clientes de la red sólo acceden a la información, además permite administrar información de otros servidores de datos, sin embargo, a pesar de contar con semejantes características no es multiplataforma, de modo que solo puede ser utilizado con el sistema operativo Windows que está patrocinado por la compañía Microsoft.
- ✓ MySQL es un sistema de gestión de base de datos relacional, multihilo y multiusuario, usado sobre todo en Internet en conjunción con PHP. Fue creado por la empresa de software libre MySQL AB y, desde enero de 2008 una subsidiaria de Sun Microsystems y ésta a su vez de Oracle Corporation desde abril de 2009— lo desarrolla como software libre en un esquema de licenciamiento dual. Por una parte MySQL se ofrece bajo la GNU GPL⁸() para cualquier uso compatible con esta licencia, pero para aquellas empresas que quieran incorporarlo en productos privativos deben comprar a la empresa una licencia específica que les permita este uso.

Al contrario de proyectos como Apache, donde el software es desarrollado por una comunidad pública y el copyright del código está en poder del autor individual, MySQL es propietario y está patrocinado por una empresa privada, que posee el copyright de la mayor parte del código.

MySQL está escrito en C y C++ y probado con multitud de compiladores y dispone de APIs para muchas plataformas diferentes. Permite manejar multitud de tipos para columnas y registros de

⁸ GNU GPL (GNU General Public License), es una licencia creada por la Free Software Foundation a mediados de los 80, y está orientada principalmente a proteger la libre distribución, modificación y uso de software.

longitud fija o variable así como que permite conexiones entre diferentes máquinas con distintos sistemas operativos.

- ✓ PostgreSQL es un gestor de base de datos de código abierto⁹ que posee una gran escalabilidad, es capaz de ajustarse al número de procesadores y a la cantidad de memoria que posee el sistema de forma óptima, haciéndole capaz de soportar una mayor cantidad de peticiones simultáneas de manera correcta.

Características de PostgreSQL:

1. Aprovecha la potencia de sistemas multiprocesador, gracias a su implementación multihilo.
2. Soporta gran cantidad de tipos de datos para las columnas.
3. Dispone de API's en gran cantidad de lenguajes, por ejemplo: C, C++, Java y PHP.
4. Gran portabilidad entre sistemas.
5. Soporta hasta 32 índices por tabla.
6. Gestión de usuarios y passwords, manteniendo un muy buen nivel de seguridad en los datos.
7. Tiene licencia de tipo BSD

Teniendo en cuenta las características analizadas de los gestores de base de datos expuestos, se opta por usar PostgreSQL en su versión 8.2, la cual incluye mejoras de rendimiento, características de SQL 2003, incluyendo funciones de agregación estadística, sentencias VALUE con múltiples registros, UPDATE RETURNING y funciones de agregación de múltiples columnas, así como índices invertidos generalizados, lo cual constituye una forma más escalable y programable de indexar datos semi-estructurados y texto.

➤ **Servidor Web. Apache**

⁹ Código abierto, es el término con el que se conoce al software distribuido y desarrollado libremente.

Un servidor Web es un programa que corre sobre el servidor escuchando las peticiones HTTP que le llegan y las satisface. Dependiendo del tipo de la petición, el servidor Web busca una página Web o bien ejecuta un programa en el servidor. De cualquier modo, siempre devuelve algún tipo de resultado HTML al cliente o navegador que realizó la petición.

Apache es un servidor altamente configurable de diseño modular que por su sencillez permite ampliar sus capacidades. Apache es una tecnología gratuita con licencia de código abierto de tipo BSD¹⁰. Además trabaja con gran cantidad de Perl, PHP y otros lenguajes de script. Funciona en Windows, en Linux y en otros sistemas de Unix.

Apache presenta muchas otras características, entre ellas un elaborado índice de directorios, un directorio de alias, negociación de contenidos, informe de errores HTTP configurable, gestión de recursos para procesos hijos, integración de imágenes del lado del servidor, reescritura de las URL, comprobación de la ortografía de las URL, y manuales online. [16]

➤ Lenguajes de Programación

Actualmente existen diferentes lenguajes de programación para desarrollar en la web, estos se han ido desarrollando 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 pasó el tiempo, las tecnologías fueron evolucionando y surgieron nuevos problemas a dar solución. Esto dio lugar a que se desarrollaran lenguajes de programación para la web, que permitieran interactuar con los usuarios y utilizaran sistemas de Bases de Datos, como es el caso de Java, Python, Perl, PHP, entre otros.

- ✓ Java es un lenguaje de programación orientado a objetos, desarrollado por una trasnacional denominada SUN Microsystem a principios de los años 90, la misma entre noviembre de 2006 y mayo de 2007, liberó la mayor parte de sus tecnologías Java bajo la licencia GNU GPL, de tal forma que prácticamente todo el Java de Sun, desarrollado hasta abril de 2009 es software libre,

¹⁰ La licencia BSD es la licencia de software otorgada principalmente para los sistemas BSD (Berkeley Software Distribution) perteneciente al grupo de licencias de software Libre.

ya que como se había mencionado anteriormente, a partir de esta fecha, Oracle Corporation compró a la empresa desarrolladora de Java.

Este lenguaje en sí mismo toma mucha de su sintaxis de C y C++, pero tiene un modelo de objetos más simple y elimina herramientas de bajo nivel, que suelen inducir a muchos errores, como la manipulación directa de punteros o memoria. Con la programación en Java, se pueden realizar distintos aplicativos, como son applets, que son aplicaciones especiales, que se ejecutan dentro de un navegador al ser cargada una página HTML en un servidor web. Java es un lenguaje seguro, robusto, con un amplio conjunto de bibliotecas estándares, es multiplataforma, de modo que el mismo código java que funciona en un sistema operativo, funciona en cualquier otro que tenga instalada la máquina virtual java, la cual tiene el inconveniente de que agrega gran complejidad al sistema en tiempo de ejecución.

- ✓ Python es un lenguaje de programación creado en el año 1990 por Guido van Rossum, es el sucesor del lenguaje de programación ABC. Los usuarios lo consideran como un lenguaje más limpio para programar. Permite la creación de todo tipo de programas incluyendo los sitios web. Su código no necesita ser compilado, por lo que se dice que el código es interpretado. Es un lenguaje de programación multiparadigma, lo cual fuerza a que los programadores adopten por un estilo de programación particular:
 - Programación orientada a objetos.
 - Programación estructurada.
 - Programación funcional.
 - Programación orientada a aspectos ¹¹(POA).

Este lenguaje es libre y de fuente abierta, de propósito general, cuenta con gran cantidad de funciones y librerías, es sencillo y rápido de programar, es multiplataforma, su licencia es de

¹¹ Programación orientada a aspectos, es un paradigma de programación cuya intención es permitir una adecuada modularización de las aplicaciones y posibilitar una mejor separación de conceptos.

código abierto (Opensource), además es portable y orientado a objetos. Es un lenguaje interpretado, lo que significa que no se necesita compilar el código fuente para poder ejecutarlo, lo que ofrece ventajas como la rapidez de desarrollo e inconvenientes como una menor velocidad. Además, a pesar de que no es un lenguaje difícil de aprender, la mayoría de su documentación se encuentra en inglés, lo que dificultaría la preparación del equipo de desarrollo.

- ✓ Perl (Lenguaje Práctico para la Extracción e Informe) es un lenguaje gratuito que toma características del C, del lenguaje interpretado shell (sh), AWK, sed, Lisp y, en un grado inferior, muchos otros lenguajes de programación. Estructuralmente, Perl está basado en un estilo de bloques como los del C o AWK, y fue ampliamente adoptado por su destreza en el procesador de texto y no tener ninguna de las limitaciones de los otros lenguajes de script.

Perl es la alternativa más popular a PHP, seguramente porque es el lenguaje más antiguo también dentro de las alternativas. Perl tiene una ventaja y es que es muy flexible, y también tiene una gran cantidad de módulos ya escritos.

Sin embargo, es lento para algunas aplicaciones y, la libertad que se le otorga al programador puede significar que el resultado sea un programa ilegible; si no se escribe con cuidado puede llegar a ser difícil de leer.

- ✓ PHP es un lenguaje de programación utilizado para la creación de sitios web. PHP es un acrónimo recursivo que significa “PHP Hypertext Pre-processor”, (inicialmente se llamó Personal Home Page). Surgió en 1995, desarrollado por PHP Group.

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 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:

- 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.
- Se necesita instalar un servidor web.

Luego del análisis de los lenguajes anteriores se decide utilizar PHP 5 para el desarrollo del sistema, ya que además de ser este un poderoso lenguaje e intérprete, ya sea incluido como parte de un servidor web en forma de módulo o ejecutado como un binario CGI (Common Gateway Interface) separado, es capaz de acceder a archivos, ejecutar comandos y abrir conexiones de red en el servidor.

PHP ha sido diseñado específicamente para ser un lenguaje más seguro para escribir programas CGI que Perl o C y, con la correcta selección de las opciones de configuración de tiempo de compilación y ejecución, se consigue la exacta combinación de libertad y seguridad que se necesita. [17]

Desde su creación en 1994, PHP ha ido incrementando sus funcionalidades debido a las contribuciones que ha recibido y que han sido evidenciadas en cada una de sus versiones. El principal objetivo de la quinta versión de PHP (PHP5) ha sido mejorar los mecanismos de Programación Orientada a Objetos (POO) para solucionar las carencias de las versiones anteriores, un paso necesario para conseguir que PHP sea un lenguaje apto para todo tipo de aplicaciones y entornos, incluso los más exigentes.

Tomando como base las tecnologías estudiadas, se pretende desarrollar una aplicación web siguiendo el paradigma de la Web 2.0, como se había mencionado anteriormente.

1.3.3 Web 2.0

La Web 2.0 representa en la actualidad la evolución que han tenido las aplicaciones web tradicionales hacia las aplicaciones Web dirigidas al usuario final. [18]

Hay que destacar que el término de la Web 2.0 no es en sí misma una tecnología, sino que es una actitud que han adoptado las nuevas aplicaciones Web en el mundo actual. Es un nuevo enfoque con el que se está desarrollando, construyendo y diseñando dichas aplicaciones en nuestros días, de manera que sean dirigidas al usuario, aumentando la interacción entre este y la aplicación.

El Sistema de Gestión “Tesis.Fac2” será elaborado y construido siguiendo este paradigma, ya que comparte las características que lo identifican, como el hecho de ser interactiva y dinámica, permitiendo una mayor participación y colaboración de los usuarios, los cuales serán consumidores y proveedores de los contenidos que se manejan. Además permite que estos productos creados o generados en este entorno se conviertan en servicios de interés para terceros, ahorra tiempo al usuario, facilita las interacciones, la publicación, la investigación y la consulta de contenidos de la web.

Para lograr este objetivo, “Tesis.Fac2” se basará en el uso de frameworks, como una tecnología para agilizar el diseño y desarrollo de aplicaciones Web.

1.3.4 Framework

El concepto framework se emplea en muchos ámbitos del desarrollo de sistemas de software. Se pueden encontrar frameworks para el desarrollo de aplicaciones médicas, de visión por computador, para el desarrollo de juegos, entre otros. En general, el término framework, se refiere a una estructura de software compuesta de componentes personalizables e intercambiables para el desarrollo de una aplicación. En otras palabras, un framework se puede considerar como una aplicación genérica incompleta y configurable a la que podemos añadirle las últimas piezas para construir una aplicación concreta.

Los objetivos principales que persigue un framework son: acelerar el proceso de desarrollo, reutilizar código ya existente y promover buenas prácticas de desarrollo como el uso de patrones.

Un framework Web, por tanto, puede ser definido como un conjunto de componentes (por ejemplo clases en java y descriptores y archivos de configuración en XML) que conforman un diseño reutilizable que facilita y agiliza el desarrollo de sistemas Web.

Un framework, en el desarrollo de software, es una estructura de soporte definida mediante la cual otro proyecto de software puede ser organizado y desarrollado. Típicamente, puede incluir soporte de programas, bibliotecas y un lenguaje interpretado entre otros softwares para ayudar a desarrollar y unir los diferentes componentes de un proyecto.

Representa una arquitectura de software que modela las relaciones generales de las entidades del dominio. Provee una estructura y una metodología de trabajo la cual extiende o utiliza las aplicaciones del dominio. [19]

Existen varios tipos de frameworks Web: orientados a la interfaz de usuario, como Java Server Faces, orientados a aplicaciones de publicación de documentos, como Cocoon, orientados a la parte de control de eventos, como Struts y algunos que incluyen varios elementos como Tapestry. La mayoría de los frameworks Web se encargan de ofrecer una capa de controladores de acuerdo con el patrón Modelo Vista Controlador (MVC) o con el modelo 2 de Servlets y JSP, ofreciendo mecanismos para facilitar la integración con otras herramientas para la implementación de las capas de negocio y presentación.

Uno de los framework más usado es Symphony, el cual está desarrollado completamente en PHP5, y se utiliza en numerosos proyectos reales como sitios Web de comercio electrónico de primer nivel. Sus propias características lo han convertido en uno de los frameworks más activos actualmente, sin embargo, tiene un alto consumo de memoria, la búsqueda de datos puede tornarse lenta cuando se desea realizarla de forma muy específica y además presenta una fuerte curva de aprendizaje, lo que podría disminuir la productividad del equipo de desarrollo durante los primeros meses, por lo que teniendo en cuenta el tiempo disponible para realizar la aplicación, se decidió utilizar CodeIgniter, el cual cumple perfectamente el fin de cualquier framework, como una excelente estructura definida para dar soporte a un proyecto web y ayudar a que este sea organizado y desarrollado.

1.3.4.1 CodeIgniter

CodeIgniter es un entorno de trabajo web para el desarrollo de aplicaciones web en PHP que facilita la escritura de código fuente en este lenguaje y reduce el tiempo necesario para lograrlo. Está basado en el patrón de arquitectura modelo vista controlador.

Una de las características más interesantes de CodeIgniter es el elevado número de clases que incluye para trabajar con distintos objetos, por ejemplo: calendario, bases de datos, correo electrónico, manipulación de imágenes, lenguaje, tablas y sesiones. Además se encuentra bajo una licencia open source Apache/BSD-style.

A diferencia de otros frameworks, CodeIgniter cuenta con una documentación excelente que permite conocer todos los secretos de este entorno de trabajo. [20]

Para poder ejecutar CodeIgniter se debe contar con un servidor web como Apache y el lenguaje de programación PHP, que fueron precisamente los seleccionados para desarrollar la aplicación. También es posible su utilización de forma remota. A continuación se muestran algunas de las razones por las que se decide escoger este framework.

- ✓ *Performance.* CodeIgniter tiene muy buen performance en comparación con Zend o CakePHP.
- ✓ *Documentación.* Tiene una muy buena documentación en línea, además de una comunidad de usuarios muy activos (hoy en día).

- ✓ *MVC*. El uso de CodeIgniter se basa en el patrón Modelo-Vista-Controlador, separando la vista, del acceso a los datos y del controlador.
- ✓ *Facilidad de uso*. La curva de aprendizaje es sumamente corta. Perfectamente se puede comenzar a utilizar las características habiendo leído el tutorial de CodeIgniter o las guías online.
- ✓ *Active Record*. El manejo y abstracción de base de datos es realmente bueno, e implementa el patrón Active Record¹².
- ✓ *Estructura organizada*. La estructura de archivos es muy clara y ofrece gran flexibilidad en caso que necesite cambiarse por algún motivo. La división en models, views, controllers, libraries, helpers, config es muy práctica y comprensible.
- ✓ *Ruteo*. Hoy en día la optimización para buscadores es fundamental en algunos casos. Los mecanismos de ruteo de CodeIgniter favorecen a éste y otras aplicaciones.

Además CodeIgniter es un poderoso Framework para PHP que facilita la escritura de código repetitivo, y a comparación de otros Frameworks cómo CakePHP, Symphony o Zend Framework, CodeIgniter es más rápido y es totalmente extensible y altamente compatible con gran variedad de versiones y configuraciones de PHP.

Sin embargo, aun cuando se cuente con una selección óptima de tecnologías y herramientas para la elaboración de una aplicación, se hace fundamental tener una guía para realizar con la calidad requerida la misma, de manera que exista una planificación del esfuerzo, tiempo, recursos, que orienten al equipo de desarrollo hacia un trabajo ordenadamente documentado y elaborado. Por esta razón se impone la necesidad de hacer uso de una metodología de desarrollo eficiente y acorde con las metas fijadas por el equipo de desarrollo.

¹² Active Record, es un patrón en el cual, el objeto contiene los datos que representan a un renglón (o registro) de una tabla o vista, además de encapsular la lógica necesaria para acceder a la base de datos. De esta forma el acceso a datos se presenta de manera uniforme a través de la aplicación.

1.4 Fundamentación de la Metodología

Desarrollar un buen software depende de un sinnúmero de actividades y etapas, donde el impacto de elegir la mejor metodología para un equipo en un determinado proyecto, es trascendental, para el éxito del producto. [21]

En un proyecto de desarrollo de software la metodología define Quién debe hacer Qué, Cuándo y Cómo debe hacerlo. Es importante señalar que no existe una metodología de software universal. Las características de cada proyecto (por ejemplo: equipo de desarrollo, recursos y tamaño del proyecto) exigen que el proceso sea configurable y determina la metodología necesaria para su realización.

El desarrollo de software no es una tarea fácil. Prueba de ello es que existen numerosas propuestas metodológicas que inciden en distintas dimensiones del proceso de desarrollo.

Por una parte se encuentran aquellas propuestas más tradicionales que se centran especialmente en el control del proceso, estableciendo rigurosamente las actividades involucradas, los artefactos que se deben producir, y las herramientas y notaciones que se usarán. [22]

Entre las principales metodologías se encuentran las muy conocidas RUP (Rational Unified Process) y MSF (Microsoft Solution Framework) entre otras, que centran su atención en llevar una documentación exhaustiva de todo el proyecto y en cumplir con un plan de proyecto, definido todo esto, en la fase inicial del desarrollo del mismo.

El **Proceso Unificado de Rational** (RUP) es un proceso de desarrollo de software, que 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.

Provee un acercamiento disciplinado para asignar tareas y responsabilidades dentro de una organización de desarrollo. Su objetivo es asegurar la producción de software de alta calidad que satisfaga los requerimientos de los usuarios finales (respetando cronograma y presupuesto). Puede ser adaptado y

extendido para satisfacer las necesidades de la organización que lo adopte. Además se caracteriza por ser iterativo e incremental, estar centrado en la arquitectura y dirigido por casos de usos¹³. [23]

En su modelación, RUP define como sus principales elementos:

Trabajadores: Definen el comportamiento y responsabilidades (rol) de un individuo, grupo de individuos, sistema automatizado o máquina, que trabajan en conjunto como un equipo. Ellos realizan las actividades y son propietarios de elementos.

Actividades: Es una tarea que tiene un propósito claro, es realizada por un trabajador y manipula elementos.

Artefactos: Productos tangibles del proyecto que son producidos, modificados y usados por las actividades. Pueden ser modelos, elementos dentro del modelo, código fuente y ejecutables.

Flujo de actividades: Secuencia de actividades realizadas por trabajadores y que produce un resultado de valor observable.

En RUP se han agrupado las actividades en grupos lógicos, definiéndose 9 flujos de trabajo principales: modelamiento del negocio, requisitos, análisis y diseño, implementación, prueba, despliegue, administración de configuración y cambios, administración del proyecto y ambiente; los 6 primeros son conocidos como flujos de ingeniería, los cuales guían todo el proceso de desarrollo del software y los tres últimos, como de apoyo, que están presentes durante todo el ciclo de vida del sistema. Además divide el proceso en cuatro fases, dentro de las cuales se realizan varias iteraciones en número variable según el proyecto y en las que se hace un mayor o menor hincapié en las distintas actividades. Estas fases son: inicio, elaboración, construcción y transición.

Por su parte, **Microsoft Solution Framework** (MSF) es un compendio de las mejores prácticas en cuanto a administración de proyectos se refiere. Más que una metodología rígida de administración de proyectos, MSF es una serie de modelos que puede adaptarse a cualquier proyecto de tecnología de información, encargados de planificar las diferentes partes implicadas en el desarrollo de un proyecto: Modelo de

¹³ Un caso de uso es un fragmento de funcionalidad del sistema que proporciona al usuario un resultado importante.

Arquitectura del Proyecto, Modelo de Equipo, Modelo de Proceso, Modelo de Gestión del Riesgo, Modelo de Diseño de Proceso y finalmente el modelo de Aplicación.

MSF se centra en los modelos de proceso y de equipo dejando en un segundo plano las elecciones tecnológicas.

Dado que es una aplicación Microsoft, tiende a una visión orientada de negocio, así que trata de agregar valor, y conserva de los últimos modelos el hecho que permite agregar cambios, busca la mayor calidad (aunque signifique mayor inversión) y trata de retroalimentarse de los aciertos y errores cometidos.

Otra aproximación es centrarse en otras dimensiones, como por ejemplo el factor humano o el producto software. Esta es la filosofía de las metodologías ágiles, las cuales dan mayor valor al individuo, a la colaboración con el cliente y al desarrollo incremental del software con iteraciones muy cortas. Este enfoque ha mostrado su efectividad en proyectos con requisitos muy cambiantes y cuando se exige reducir drásticamente los tiempos de desarrollo pero manteniendo una alta calidad.

Entre los principales métodos ágiles se encuentran XP (eXtreme Programming), Scrum, Iconix, Cristal Methods, AUP entre otras. Estas metodologías tienen como característica que la capacidad de respuesta a un cambio es más importante que el seguimiento estricto de un plan.

La Programación Extrema (XP) es una metodología ligera de software que se basa en la simplicidad, la comunicación y la realimentación o reutilización del código desarrollado. Tiene como objetivos dar al cliente el software que él necesita y cuando lo necesite, por lo que se debe responder a los cambios inmediatamente, incluso aún cuando estos sean dados al final del ciclo de la programación y potenciar al máximo el trabajo en grupo, de modo que tanto los jefes de proyecto, los clientes y desarrolladores, sean parte del equipo y estén involucrados en el desarrollo del software.

XP define cuatro variables para proyectos de software:

- ✓ Coste: Máquinas, especialistas y oficinas.
- ✓ Tiempo: Total y de Entregas.
- ✓ Calidad: Externa e Interna.

✓ Alcance: Intervención del cliente.

La agilidad de esta metodología la hace especialmente útil en un entorno de continuas redefiniciones en los requisitos. Se plantea como una metodología a emplear en proyectos de riesgo.

Metodología Seleccionada. RUP:

Luego de analizar y comparar estas metodologías se decide utilizar RUP para el desarrollo de la aplicación, teniendo en cuenta las características de la misma, ya que se persigue lograr un sistema con la robustez necesaria, independientemente del plazo fijado para su entrega y el tamaño del equipo, anteponiendo sobre todo, la precisión y calidad en el sistema, debido a la importancia de los datos que se manejan, así como garantizar la documentación del mismo.

Para lograr estos objetivos es práctico dividir el trabajo en partes más pequeñas o miniproyectos. Cada miniproyecto es una iteración que resulta en un incremento. Las iteraciones hacen referencia a pasos en los flujos de trabajo, y los incrementos, al crecimiento del producto. [23]

Las metodologías de desarrollo, tienen entre sus objetivos mejorar la calidad de los proyectos realizados y aumentar la productividad, sin embargo aún cuando se tenga una buena organización y se haya seleccionado una buena metodología, es necesario el uso de herramientas que auxilien este trabajo.

1.4.1 Herramientas Case

Las Herramientas CASE (Computer-Aided Systems Engineering) se pueden definir como un conjunto de programas y ayudas que dan asistencia a los analistas, ingenieros de software y desarrolladores, durante todos los pasos del ciclo de vida de desarrollo de un software. Existen herramientas CASE de trabajo visuales como el Rational Rose y el Visual Paradigm, entre otras que permiten realizar el modelado del desarrollo de los proyectos.

El *Rational Rose* provee el modelado basado en UML (Unified Modeling Language). La Corporación Rational ofrece un Proceso Unificado (RUP) para el desarrollo de los proyectos de software, desde la etapa de Ingeniería de Requerimientos hasta la de pruebas. Para cada una de estas etapas existe una herramienta de ayuda en la administración de los proyectos, Rose es la herramienta del Rational para la etapa de análisis y diseño de sistemas. Rose es una herramienta con plataforma independiente que ayuda

a la comunicación entre los miembros de equipo, a monitorear el tiempo de desarrollo y a entender el entorno de los sistemas. Una de sus grandes ventajas es que utiliza la notación estándar en la arquitectura de software (UML), la cual permite a los arquitectos de software y desarrolladores visualizar el sistema completo utilizando un lenguaje común, además los diseñadores pueden modelar sus componentes e interfaces en forma individual y luego unirlos con otros componentes del proyecto.

También fue objeto de estudio Visual Paradigm, la cual es una herramienta que interpreta UML profesional, que soporta el ciclo de vida completo del desarrollo de software: análisis y diseño orientados a objetos, construcción, pruebas y despliegue. Este software 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. Una de las características más importantes del Visual Paradigm es que es multiplataforma.

Se integra con Eclipse/IBM WebSphere, Borland JBuilder, NetBeans IDE/Sun ONE, IntelliJ IDEA, Oracle JDeveloper, BEA Weblogic Workshop y ofrece:

- ✓ Entorno de creación de diagramas para UML 2.0 y 2.1.
- ✓ Diseño centrado en casos de uso y enfocado al negocio que generan un software de mayor calidad.
- ✓ Uso de un lenguaje estándar común a todo el equipo de desarrollo que facilita la comunicación.
- ✓ Capacidades de ingeniería directa (versión profesional) e inversa.
- ✓ Modelo y código que permanece sincronizado en todo el ciclo de desarrollo.
- ✓ Disponibilidad de múltiples versiones, para cada necesidad.
- ✓ Disponibilidad de integrarse en los principales IDEs¹⁴.

¹⁴ IDEs, es un entorno de programación que ha sido empaquetado como un programa de aplicación, es decir, consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica.

- ✓ Disponibilidad en múltiples plataformas como: Microsoft Windows (98, 2000, XP, o Vista), Linux y Mac OS X.
- ✓ Distribución automática de diagramas, reorganización de las figuras y conectores de los diagramas UML.
- ✓ Interoperabilidad con otras aplicaciones, ya que brinda la posibilidad de intercambiar información mediante la importación y exportación de ficheros con aplicaciones como por ejemplo Visio y Rational Rose.
- ✓ Generación de código ORM (Object-Relational Map), pues permite generar, a partir de un Diagrama de Entidad Relación, una Base de Datos Relacional y el código necesario para acceder a esta base de datos utilizando PHP entre otros lenguajes.

Independientemente de que la herramienta Rational haya sido pensada sobre la base de un proceso unificado como lo es RUP y a pesar de haber sido esta la metodología seleccionada para desarrollar este trabajo, se ha decidido utilizar Visual Paradigm como herramienta CASE, debido fundamentalmente a que es multiplataforma y que la UCI tiene comprada su licencia.

1.5 Conclusiones

En este capítulo se realizó un análisis del tema de la investigación a nivel internacional, además de una investigación de las principales metodologías de desarrollo de software, con el fin de justificar la empleada teniendo en cuenta las características del sistema a desarrollar. Por otra parte se realizó un estudio de las principales herramientas y tecnologías a utilizar para la realización de este trabajo teniendo en cuenta criterios que hagan óptimo al sistema.

Capítulo 2. Características del sistema.

2.1 Introducción

Satisfacer las necesidades del cliente es el objetivo fundamental de todo producto de software, para ello es necesario conocer cuál es el problema y la situación problemática que se debe solucionar, así como el objeto de automatización y la información que se maneja. La comprensión de lo planteado será la base para identificar: actores, trabajadores y casos de uso del negocio y para capturar los requisitos funcionales y no funcionales de la aplicación a implementar, a partir de los cuales se seleccionarán los actores y casos de uso del sistema. Este capítulo recoge los elementos comentados anteriormente para la propuesta de construcción de la aplicación informática a desarrollar.

2.2 Objeto de estudio

2.2.1 Problema y situación problemática

En la Facultad 2 existen actualmente dificultades con la asignación y control de las tesis a los estudiantes de quinto año, ya que no se cuenta con una estructura diseñada específicamente para manejar este proceso, de modo que se agilice el trabajo tanto del personal involucrado como de los tesisistas.

2.2.2 Objeto de automatización.

Durante el etapa de desarrollo de las tesis, desde la elaboración del perfil hasta la aprobación y discusión final existen varios procesos que se realizan hasta el momento manualmente y que deben ser automatizados, ya que su manipulación resulta compleja y está propensa a errores, que pueden incidir negativamente en la calidad del trabajo final, además que consume tiempo al personal encargado de realizar las tareas. Con el presente trabajo se pretende desarrollar una aplicación capaz de automatizar todo el proceso que involucra las tesis de la Facultad 2. Se espera que el mismo unifique toda la información requerida tanto por los profesores como por los estudiantes, así como brindar los reportes necesarios para mantener un conocimiento general del estado de las tesis y del personal involucrado.

2.2.3 Información que se maneja.

La información que se maneja principalmente se resume en los documentos de perfil de tesis, los cuales tienen un formato específico, los documentos de tesis, que en su generalidad tienen un volumen considerable de información compendiada, referente a las aplicaciones o investigaciones desarrolladas por los tesisistas, los resultados de los cortes de tesis, actas de pre defensa y defensa, opiniones del tutor y el oponente, así como los tribunales y las comisiones de tesis, además de otros documentos de carácter oficial o no que pueda publicar el personal autorizado, con fines informativos u otro.

2.3 Propuesta del sistema

Para dar solución al problema existente se ha decidido desarrollar un sistema de gestión que facilite el trabajo con la información referente a las tesis y que garantice una mejor coordinación de las mismas. Para esto se implementará una aplicación web siguiendo el paradigma de la Web 2.0 haciendo uso de CodeIgniter como framework para el desarrollo de aplicaciones web en PHP.

El trabajo estará basado en algunos procesos fundamentales como el de la aprobación de los perfiles de tesis, para lo cual se requiere de una gestión eficiente de los temas, por parte de los líderes de proyectos y de los miembros del Comité de tesis de la Facultad 2. Otro proceso importante es el seguimiento y control de los trabajos desarrollados por cada equipo, que será garantizado por los tutores, líderes y miembros del tribunal al que esta pertenezca y finalmente los reportes y boletines informativos que la aplicación debe ser capaz de brindar, de modo que sirva como vía de comunicación para obtener la información correcta y actualizada de los eventos a realizarse.

2.4 Modelo de negocio.

El primer paso hacia la construcción plena de la aplicación es desarrollar el modelo de negocio, el cual describe los procesos de negocio, identificando quiénes participan y las actividades que requieren automatización. Este modelo se realiza con el objetivo de comprender la estructura y la dinámica de la organización en la cual se va a implantar el sistema, en este caso la Facultad 2, comprender los problemas actuales de la misma e identificar las mejoras potenciales y asegurar que los consumidores, usuarios finales y desarrolladores tengan un entendimiento común de la organización.

2.4.1 Descripción General del Negocio.

El Comité de Tesis de la Facultad 2 está conformado por un amplio conjunto de profesores y directivos que son los encargados de garantizar la distribución de los temas de tesis a cada estudiante de quinto año. Para ello deben tener en cuenta quiénes de ellos están vinculados a proyectos o pertenecen a las FAR o al MININT y por tanto sus trabajos de tesis se desarrollan por sus respectivas áreas de trabajo. A los que no están involucrados en ninguno de estos dos casos excepcionales se les proponen temas que incluyen el desarrollo de aplicaciones o que pueden estar relacionados con temas investigativos que son del interés de algún profesor, de la Facultad o de la Universidad. Luego a cada equipo de tesis, que puede estar conformado como máximo por dos estudiantes, se le asigna un tema de tesis, un tutor y, un cotutor en caso de ser preciso. Estos elaboran el perfil de tesis, que debe ser aprobado por el Comité de Tesis de la Facultad. Toda esta estructura, durante el transcurso del tiempo puede sufrir modificaciones, que indudablemente tienden a retrasar el trabajo, aunque se hace lo posible para que todo ocurra durante la primera etapa del curso.

Se cuenta con un período de aproximadamente 9 meses para la terminación total del trabajo, durante el cual se realizan cortes para lograr una supervisión del mismo y de ese modo llegar a la exposición final con un mínimo de errores. Para esto se conforman comisiones de tesis, compuestas por profesores afines con el tema, los cuales en cada corte emiten una evaluación de la misma. Los tutores por su parte deben supervisar el trabajo sistemáticamente, haciendo las sugerencias pertinentes para mejorar su calidad.

2.4.2 Actores del Negocio

Un actor del negocio es cualquier individuo, grupo, organización o máquina que interactúa con el negocio [24].

Actores	Descripción
Estudiante	Persona que asume el desarrollo de un tema de tesis asignado por el comité de tesis de la facultad o por su líder de proyecto.

Tabla 1: Relación de actores del negocio

2.4.3 Trabajadores del Negocio

Un trabajador del negocio representa a personas, o sistemas (software) dentro del negocio que son las que realizan las actividades que están comprendidas dentro de un caso de uso.

Trabajadores	Descripción
Tutor	Persona encargada de realizar el perfil de tesis y de supervisar la evolución de las tesis desarrolladas por los estudiantes.
Comité de Tesis	Encargado de proponer los temas de tesis a los estudiantes que no pertenecen a ningún proyecto, conformar los tribunales, planificar los cortes, pre defensa y defensa final, así como de garantizar que cada tesis sea asistida por al menos un tutor.
Líder de Proyecto	Encargado de asignar temas de tesis y tutores a los estudiantes que se encuentran en su proyecto.
Miembro de Tribunal	Persona encargada de la evaluar las tesis en cada corte, pre defensa y defensa final.

Tabla 2: Relación de trabajadores del negocio.

2.4.4 Diagrama de Casos de Uso del Negocio

Un caso de uso del negocio representa a un proceso de negocio, por lo que se corresponde con una secuencia de acciones que producen un resultado observable para ciertos actores del negocio.

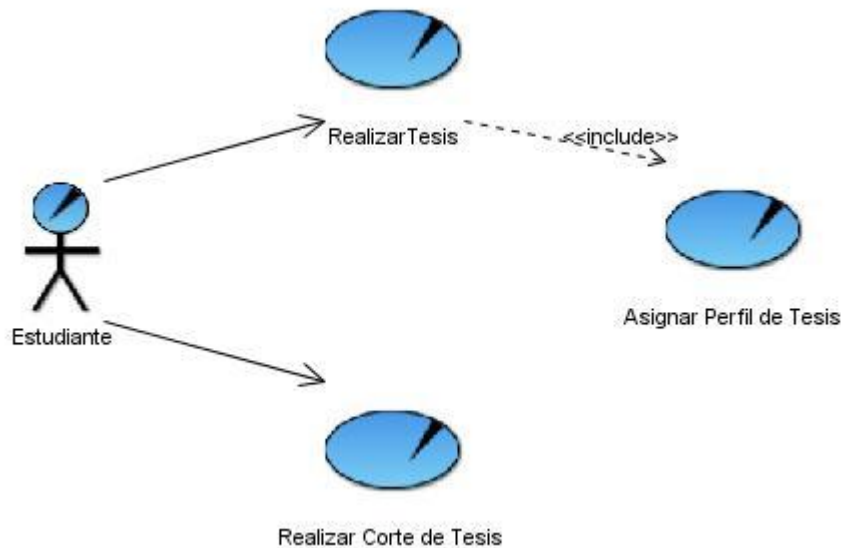


Figura 3: Diagrama de Casos de Uso del Negocio

2.4.5 Descripción textual de los Casos de Uso del Negocio. [Ver anexo 1](#)

2.4.6 Diagramas de Actividades. [Ver anexo 2](#)

2.4.7 Modelo de Objetos. [Ver anexo 3](#)

2.5 Relación de requerimientos.

Los requerimientos definen qué es lo que el sistema debe hacer, para lo cual se identifican las funcionalidades requeridas y las restricciones que se imponen.

2.5.1 Requerimientos funcionales.

Los requerimientos funcionales son capacidades o condiciones que el sistema debe cumplir, en este caso la aplicación que se pretende desarrollar debe permitir:

1. Autenticar usuarios.
2. Controlar información relacionada con los proyectos
 - 2.1 Adicionar nuevo proyecto.

- 2.1.1 Información que se maneja: Nombre del proyecto, líder del proyecto, estudiantes que integran el proyecto
- 2.2 Mostrar datos de proyectos.
- 2.3 Modificar datos de un proyecto.
- 2.4 Eliminar proyecto.
- 3. Proponer tema de tesis.
 - 3.1 Crear propuesta de tesis.
 - 3.2 Modificar datos de la propuesta de tesis.
 - 3.3 Eliminar propuesta de tesis.
 - 3.4 Mostrar propuesta de tesis.
- 4. Evaluar perfil de tesis.
 - 4.1 Aprobar perfil de tesis.
 - 4.2 Rechazar perfil de tesis.
- 5. Controlar información de los tribunales de tesis.
 - 5.1 Crear tribunal de tesis.
 - 5.1.1 Información que se maneja: Clasificación del tribunal, nombre del tribunal, presidente del tribunal, tesis que serán evaluadas en el tribunal.
 - 5.2 Mostrar tribunal de tesis.
 - 5.3 Modificar tribunal de tesis.
 - 5.4 Eliminar tribunal de tesis.
- 6. Controlar información de los cortes de tesis.
 - 6.1 Crear corte de tesis.
 - 6.1.1 Información que se maneja: Nombre del corte, lugar y fecha de realización, objetivo a evaluar, nombre del tribunal que sesionará en dicho corte.
 - 6.2 Modificar corte de tesis.
 - 6.3 Eliminar corte de tesis.
 - 6.4 Mostrar corte de tesis.
- 7. Controlar perfil de tesis.
 - 7.1 Crear perfil de tesis.

- 7.1.1 Información que se maneja: curso docente, nombre de la tesis, clasificación, autor(es), tutor, objetivo general, objetivos específicos, problema a resolver, posibles resultados, síntesis del tutor.
- 7.2 Mostrar perfil de tesis.
- 7.3 Editar perfil de tesis.
- 7.4 Eliminar perfil de tesis.
- 7.5 Notificar información del perfil.
- 8. Controlar documento de tesis.
 - 8.1 Archivar documento de tesis.
 - 8.2 Descargar documento de tesis.
 - 8.3 Eliminar documento de tesis.
 - 8.4 Listar documentos archivados.
- 9. Controlar información de los roles de usuario.
 - 9.1 Información que se maneja: nombre del rol, permisos.
- 10. Controlar información de usuarios.
 - 10.1 Crear usuario.
 - 10.1.1 Información que se maneja: nombre y apellidos del usuario, solapín, login.
 - 10.2 Mostrar usuarios.
 - 10.3 Editar usuario.
 - 10.4 Eliminar usuario.
- 11. Controlar notas de los cortes por tesis.
 - 11.1 Emitir nota de los cortes por tesis.
 - 11.2 Mostrar nota de los cortes por tesis.
 - 11.3 Cambiar nota de los cortes por tesis.
 - 11.4 Calcular nota final.
- 12. Controlar información de los comentarios.
 - 12.1 Realizar comentario.
 - 12.1.1 Información que se maneja: nombre y apellidos del autor, fecha de realización del comentario, contenido del comentario.
 - 12.2 Mostrar comentario.

12.3 Eliminar comentario.

13. Realizar reportes.

- Estudiantes con una evaluación determinada en un corte dado.
- Listado tutores
- Listado de tesis.
- Listado de profesores sin tesis.
- Listado de tesis de un tribunal determinado.
- Listado de presidentes de los tribunales

14. Controlar información de las versiones de los documentos.

14.1 Información que se maneja: nombre del documento, fecha, dirección, tesis a la que pertenece.

15. Publicar información.

15.1 Crear boletines informativos.

15.2 Mostrar boletines informativos.

15.3 Modificar boletines informativos.

15.4 Eliminar boletines informativos.

16. Emitir informe general por cortes de tesis.

17. Controlar reclamaciones de notas de cortes de tesis.

17.1 Crear reclamación.

17.2 Enviar reclamación.

17.3 Mostrar reclamación.

18. Responder reclamación de notas de cortes de tesis.

18.1 Agregar respuesta.

18.2 Enviar respuesta.

18.3 Mostrar respuesta.

2.5.2 Requerimientos no funcionales

Los requerimientos no funcionales son propiedades o cualidades que el producto debe tener, y son los que harán de la aplicación un producto atractivo, usable, rápido y confiable.

1. Apariencia o interfaz externa:

- Contará con una interfaz sencilla y con colores agradables a la vista como los distintos tonos de azules, blancos y algunas tonalidades de gris. Hará uso de banners discretos y un mapa de navegación cómodo para el usuario.

2. Usabilidad:

- El sistema podrá ser usado de forma fácil por cualquier persona, aunque será utilizado en su mayoría por los estudiantes de quinto año y los profesores involucrados en el proceso de tutoría y evaluación de tesis.

3. Rendimiento:

- El sistema será implementado bajo el paradigma de la web 2.0 que establece como característica fundamental la interactividad entre la aplicación y el usuario, de este modo, el tiempo de respuesta debe ser el menor posible, haciendo uso a la vez de páginas dinámicas que permitan un rápido acceso a la información y de la manera más fácil posible.
- El sistema contará con una base de datos en 3era forma normal que garantiza el rendimiento óptimo de la misma.
- La aplicación permitirá que múltiples usuarios estén conectados a la vez.

4. Soporte:

- El sistema será probado, instalado y configurado por los miembros del Comité de Tesis y un Administrador, el cual se ocupará también de su mantenimiento.
- El producto requiere de constantes actualizaciones debido a los cambios que pueden ocurrir respecto al personal, estudiantes, proyectos y otras informaciones que se manejan y que son de vital importancia para el correcto funcionamiento del sistema.

5. Portabilidad:

- Este producto podrá ser utilizado tanto en Windows como en Linux, ya que en su implementación se utilizarán herramientas multiplataformas.

6. Seguridad:

Dado que el sistema se pondrá en práctica dentro de la Universidad de las Ciencias Informáticas, sus mecanismos de seguridad estarán en perfecta consonancia con las políticas de seguridad establecidas en la universidad para garantizar la confidencialidad, integridad y disponibilidad de los datos que se manejan.

- *Confidencialidad:* La información que se maneja estará protegida de acceso no autorizado, ya que será requerida la autenticación de los usuarios para garantizar que solo las personas autorizadas puedan acceder a estos datos.

- *Integridad:* Los documentos e información privada de cada tesis serán objeto de cuidadosa protección contra la corrupción y estados de inconsistencia, ya que cada equipo de tesis tendrá a su disposición un espacio para almacenar documentos, que solo podrá ser accedido por el personal asociado a la tesis y los estudiantes miembros de dicho equipo, los cuales serán los únicos que tendrán la posibilidad de eliminar sus documentos. De igual modo, solamente los usuarios miembros del Comité de Tesis, podrán administrar la información referente a cortes de tesis, tribunales, evaluaciones de perfiles y clasificaciones de tesis.

- *Disponibilidad:* La aplicación estará disponible en todo momento para aquellas personas con acceso a la información, y los mecanismos utilizados para lograr la seguridad no serán un obstáculo a los usuarios para obtener los datos deseados en el momento que lo requieran.

7. Confiabilidad:

- La información almacenada, procesada y generada por el sistema será confiable, ya que a las carpetas donde se almacenan los documentos, así como a la base de datos, se le harán copias de seguridad diarias como medida preventiva ante los fallos que pudieran ocurrir.

8. Software:

- Sistema Operativo Linux, Windows 95 ó superior.
- Servidor Web Apache.
- Gestor de Base de Datos Postgre SQL 8.2.

- Framework de desarrollo: CodeIgniter.
- Debe estar habilitada en el navegador la opción de JavaScript.
- Navegador: Mozilla Firefox

9. Hardware:

- Para la PC cliente se requiere una máquina con 128 MB de RAM como mínimo, el servidor Web, al igual que el servidor de base de datos, debe tener 512 MB de RAM y 60 GB de disco duro mínimo, todas las máquinas implicadas en la funcionalidad de la aplicación deben estar conectadas a la red de al menos 100 Mbps de velocidad.

10. Políticos-culturales:

- El sistema estará completamente identificado con los principales conceptos del campo de trabajo, para que el usuario lo asocie con la vida real. El mismo responderá a los principios e intereses de la sociedad y de la Universidad.

11. Ayuda y documentación en línea:

- El sistema contará con mensajes de ayuda que indiquen qué operación realiza cada componente.

2.6 Modelo de Casos de Uso del Sistema.

El modelo de casos de uso (CU) describe la funcionalidad propuesta del nuevo sistema. Un caso de uso es un artefacto narrativo que describe, bajo la forma de acciones y reacciones, el comportamiento del sistema desde el punto de vista del usuario.

2.6.1 Definición de los actores del sistema a automatizar

Un actor del sistema es aquella persona o sistema que interactúa con el sistema.

Actores	Descripción
Estudiante	Persona que accede al sistema para ver su perfil de tesis y desarrolla su trabajo de diploma. Este actor se informa sobre la planificación de los cortes, pre defensa, defensa, y puede hacer

	reclamaciones de sus notas y otros eventos afines.
Tutor	Persona que accede al sistema para elaborar el perfil y revisar el documento de la tesis que él mismo asiste, así como realizar los señalamientos pertinentes.
Comité de Tesis	Persona que accede al sistema para publicar información, gestionar el personal de cada comisión de tesis y emitir los informes de cada corte, así como gestionar los temas, tutores y perfiles de las tesis asignadas.
Miembro del tribunal	Persona que accede al sistema para emitir las notas de cada corte y leer los documentos de las tesis de su tribunal, así como responder a las reclamaciones realizadas por parte de los estudiantes.
Líder de proyecto	Persona que accede al sistema para gestionar los temas de tesis y los tutores que asistirán las tesis de su proyecto, así como agregar comentarios sobre las mismas.
Administrador	Persona encargada de administrar el sistema, delegar permisos y gestionar los roles de cada usuario.

Tabla 3: Relación de actores del sistema

2.6.2 Diagrama de casos de uso del sistema a automatizar.

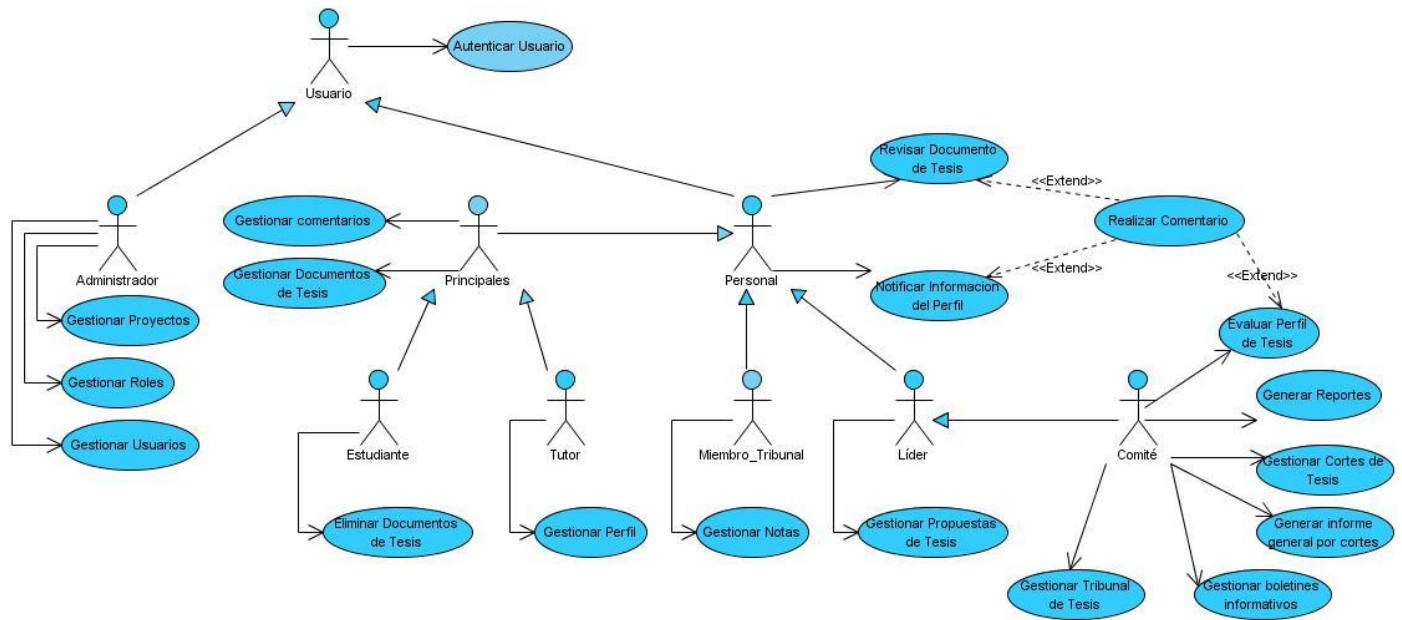


Figura 4: Diagrama de Casos de Uso del Sistema

2.6.3 Patrones de casos de uso

Un patrón es una pareja de problema / solución con un nombre, que codifica (estandariza) buenos principios y sugerencias relacionados frecuentemente con la asignación de responsabilidades.

Para desarrollar el presente modelo de casos de uso, se aplicaron patrones que permitieron reflejar con más precisión los requisitos reales y lograr mejores resultados de forma más rápida, haciendo más fácil el trabajo, los mismos se muestran a continuación:

Concordancia: Se aplicó para extraer una subsecuencia de acciones que aparecen en diferentes lugares del flujo de casos de uso y que es expresado por separado. Específicamente se realizó una especialización de casos de uso de tipo de uso común, en la que todas las acciones son heredadas por los casos de uso hijos, y otras serán adicionadas o acciones heredadas que serán especializadas.

CRUD (Crear, Obtener, Actualizar, Eliminar): Este patrón se usó para fusionar casos de uso simples, formando una unidad conceptual. Específicamente se utilizó CRUD completo para modelar todas las

operaciones que pueden ser realizadas sobre una parte de información de un tipo determinado; tales como creación, lectura, actualización y eliminación, y además CRUD parcial, ya que la acción de eliminar documentos solo la puede realizar el estudiante, y por tanto el resto de las funcionalidades propias de un gestor se encuentran separadas para que el tutor pueda acceder a ellas.

Múltiples actores (Roles comunes): Este patrón se utilizó para representar actores que interpretan el mismo papel en determinado caso de uso. Este rol es representado por otro actor, que contiene de forma hereditaria los actores que comparten este rol.

La aplicación de estos patrones permitieron resolver de manera ágil, problemas que se presentaron en la modelación del sistema, obteniendo modelos de mayor calidad de forma más rápida.

2.6.4 Descripción textual de los Casos de Uso del Sistema. [Ver anexo 4](#)

2.7 Conclusiones

En el presente capítulo se analizaron las características fundamentales del sistema. En tal sentido y a partir del análisis de los procesos de negocio se obtuvo un listado con los requerimientos funcionales y no funcionales para el desarrollo del sistema, que se representaron en el Diagrama de Casos de Uso del Sistema y finalmente se describieron todas las acciones de los actores del sistema dentro de los procesos con los que se vinculan.

Capítulo 3. Análisis y diseño del sistema

3.1 Introducción

En el presente capítulo se realiza inicialmente un análisis detallado de la solución propuesta, con el objetivo de conseguir una comprensión más precisa de los requisitos funcionales definidos anteriormente y proporcionar una visión general del sistema, para luego, a través del diseño, modelarlo de forma que cumpla con todos los objetivos trazados. Para ello se describen las clases involucradas, así como los diagramas necesarios para un mejor entendimiento de la aplicación.

3.2 Definición de diagrama de clases

Un diagrama de clases es un tipo de diagrama estático que describe la estructura de un sistema mostrando sus clases, atributos y las relaciones entre ellos. Estos son utilizados durante el proceso de análisis y diseño de los sistemas, donde se crea el diseño conceptual de la información que se manejará, y los componentes que se encargarán del funcionamiento y la relación entre uno y otro. Los diagramas de clases son el pilar básico del modelado con UML, siendo utilizados tanto para mostrar lo que el sistema puede hacer (análisis), como para mostrar cómo puede ser construido (diseño). Durante el análisis, el diagrama se desarrolla buscando una solución ideal y durante el diseño, se usa el mismo diagrama y se modifica para satisfacer los detalles de las implementaciones.

3.3 Análisis del sistema

El análisis se realiza durante la fase de elaboración y consiste en obtener una visión del sistema que se preocupa de ver QUÉ hace.

3.3.1 Modelo de Análisis

El Modelo de Análisis es el resultado de la actividad de analizar los casos de uso. Para su construcción se identificaron las clases que describen la realización de los casos de uso, los atributos y las relaciones entre ellas y con esta información se construyó el Diagrama de clases del análisis. El modelo de análisis que se muestra a continuación constituyó la primera representación técnica del sistema.

3.3.2 Diagramas de clases del análisis

El diagrama de clases del análisis es un artefacto en el que se representan los conceptos en un dominio del problema. En este diagrama se representan las clases y sus relaciones y expresan una vista estática del sistema.

A continuación se presentan los Diagramas de Clases del Análisis (DCA) por cada caso de uso del sistema (CUS):

Caso de Uso: “Autenticar Usuario”

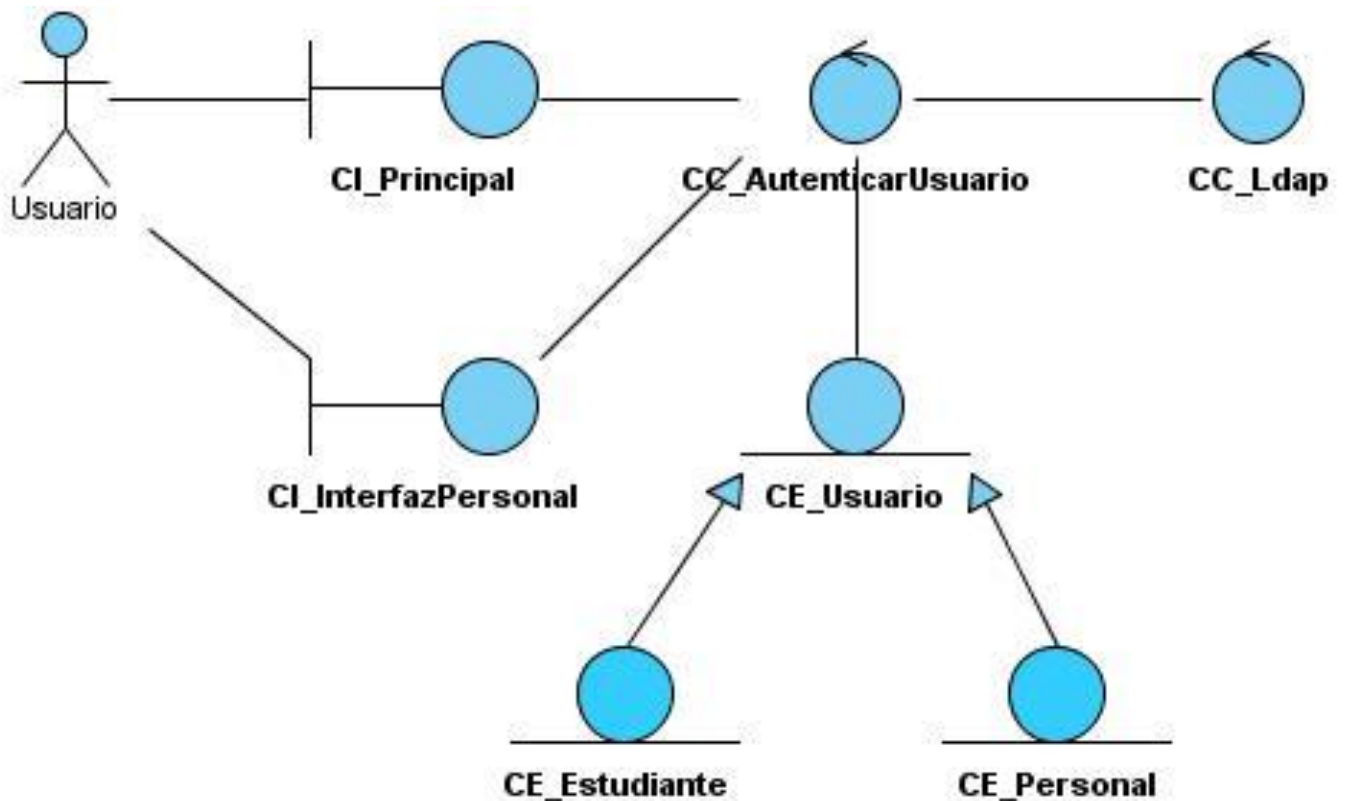


Figura 5: Diagrama de Clases del Análisis del CUS “Autenticar Usuario”

Caso de Uso: "Evaluar Perfil"

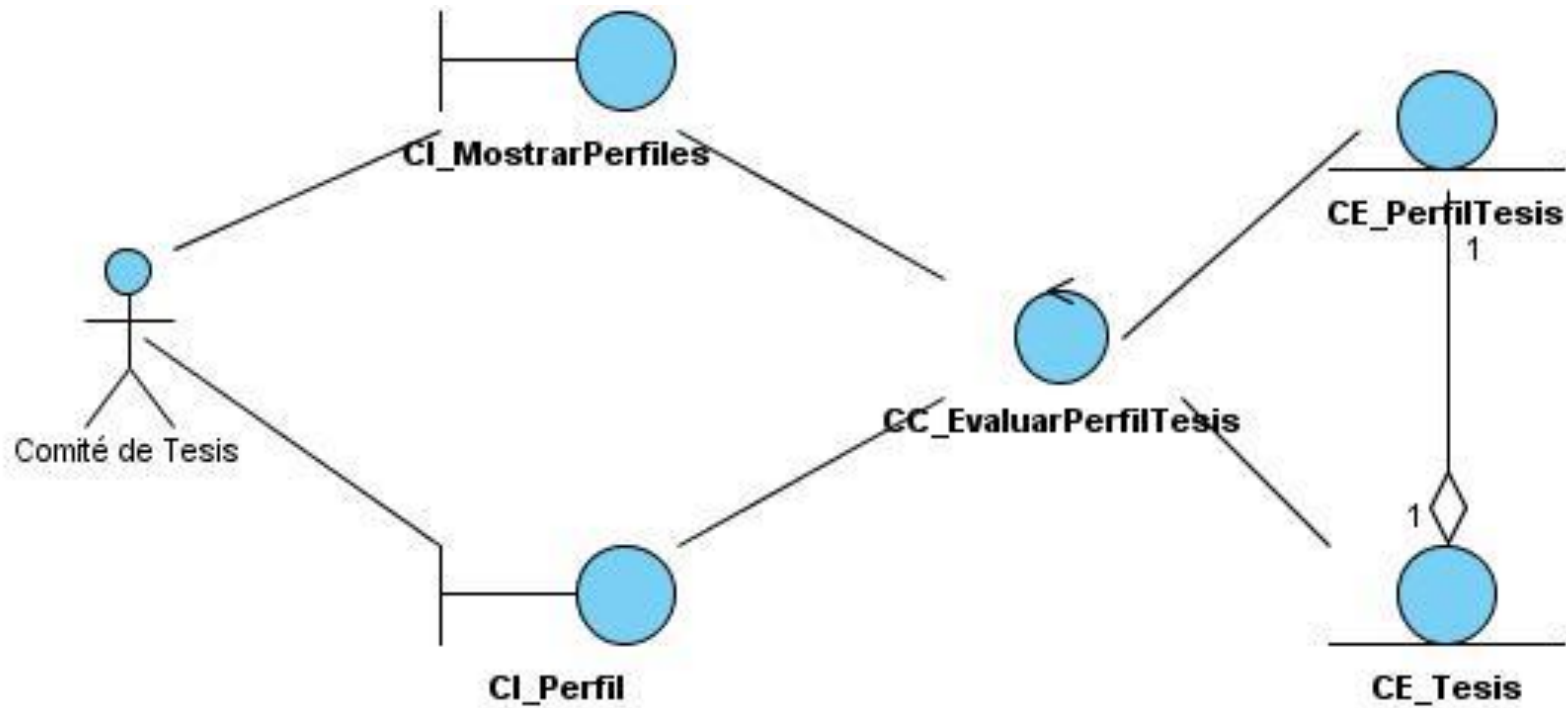


Figura 6: Diagrama de Clases del Análisis del CUS "Evaluar Perfil"

Caso de Uso: "Gestionar Perfil de Tesis"

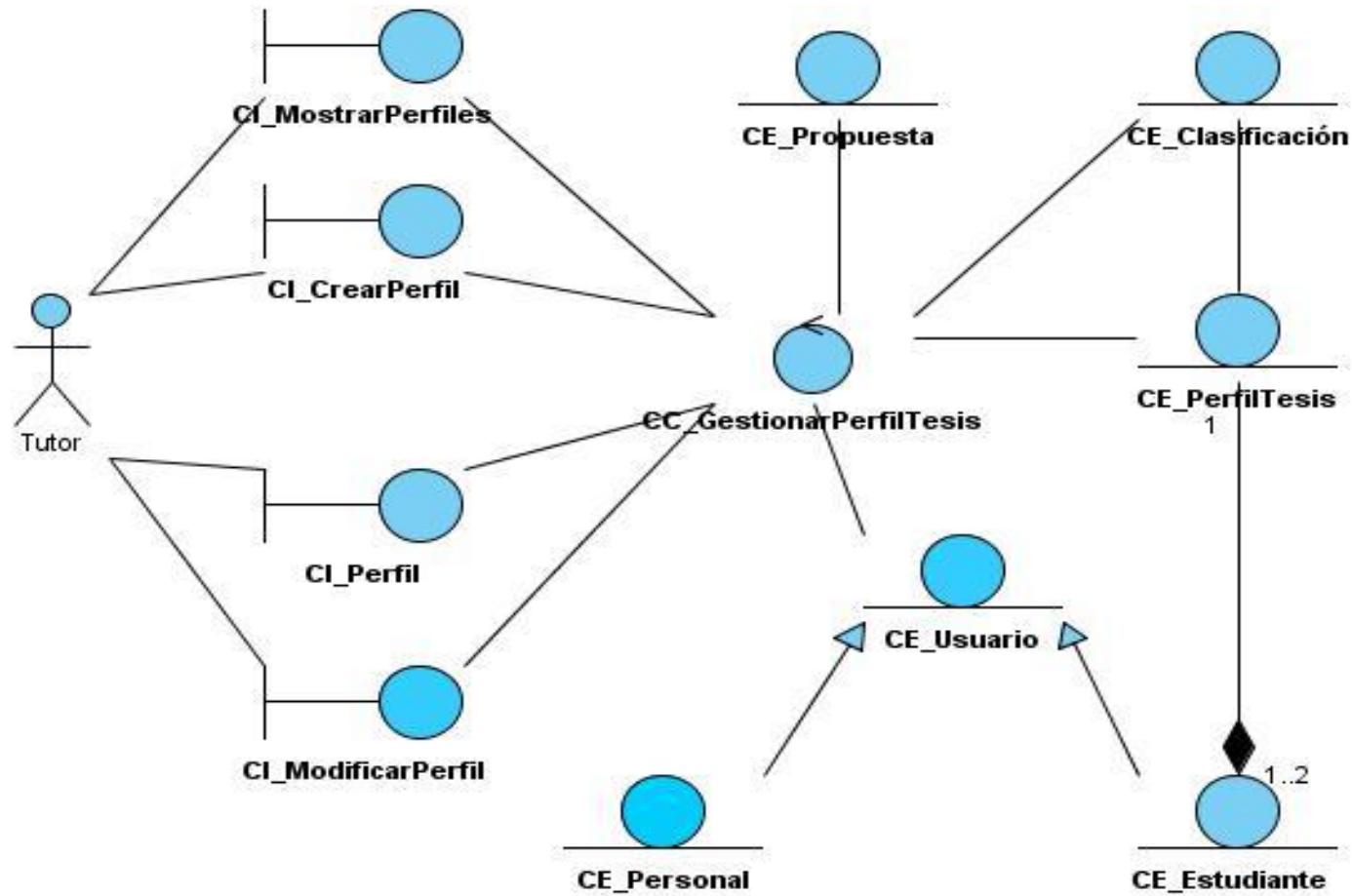


Figura 7: Diagrama de Clases del Análisis del CUS "Gestionar Perfil de Tesis"

Caso de Uso: "Gestionar Cortes de Tesis"

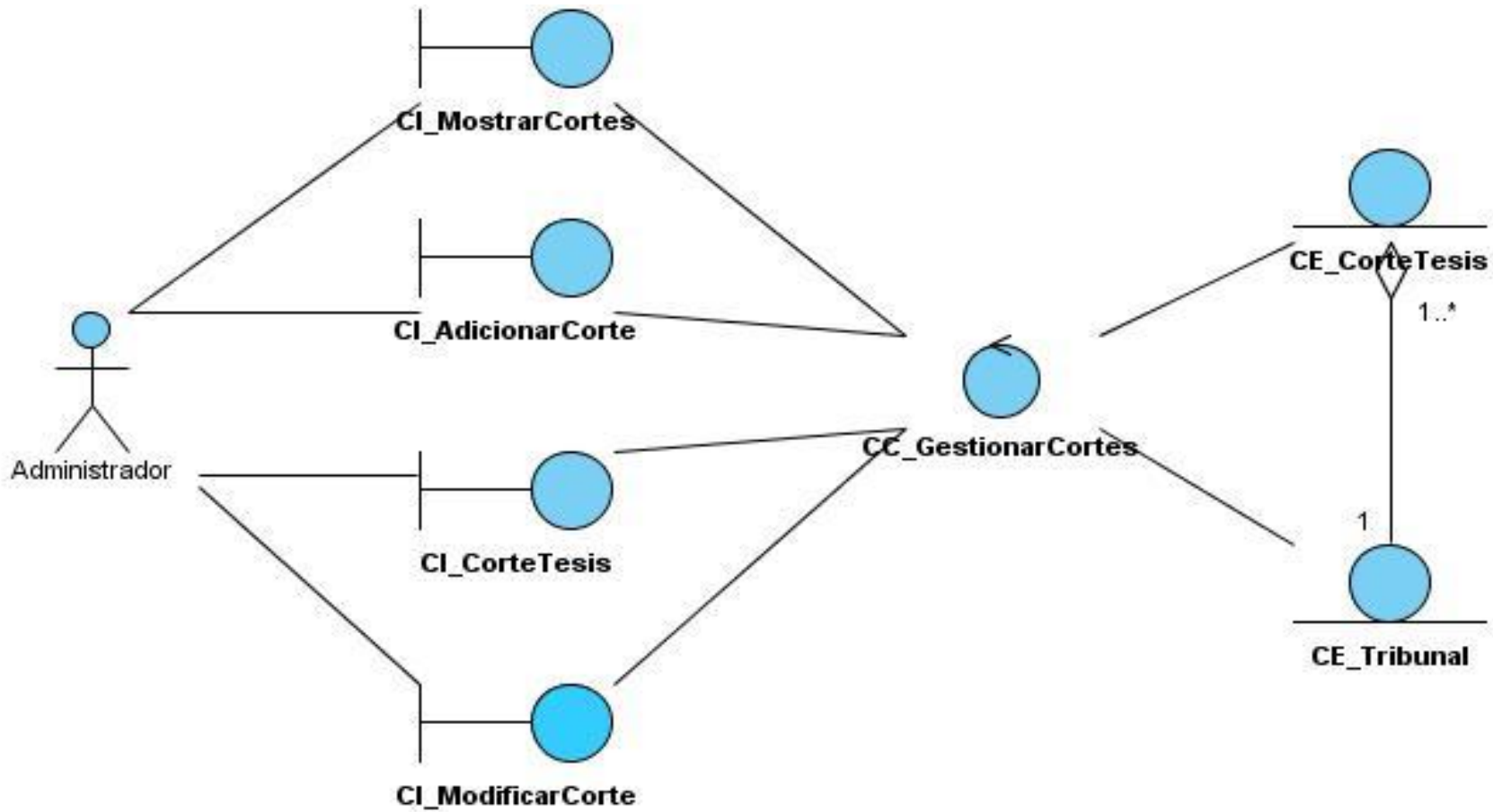


Figura 8: Diagrama de Clases del Análisis del CUS "Gestionar Cortes de Tesis"

Caso de Uso: "Gestionar Documentos de Tesis"

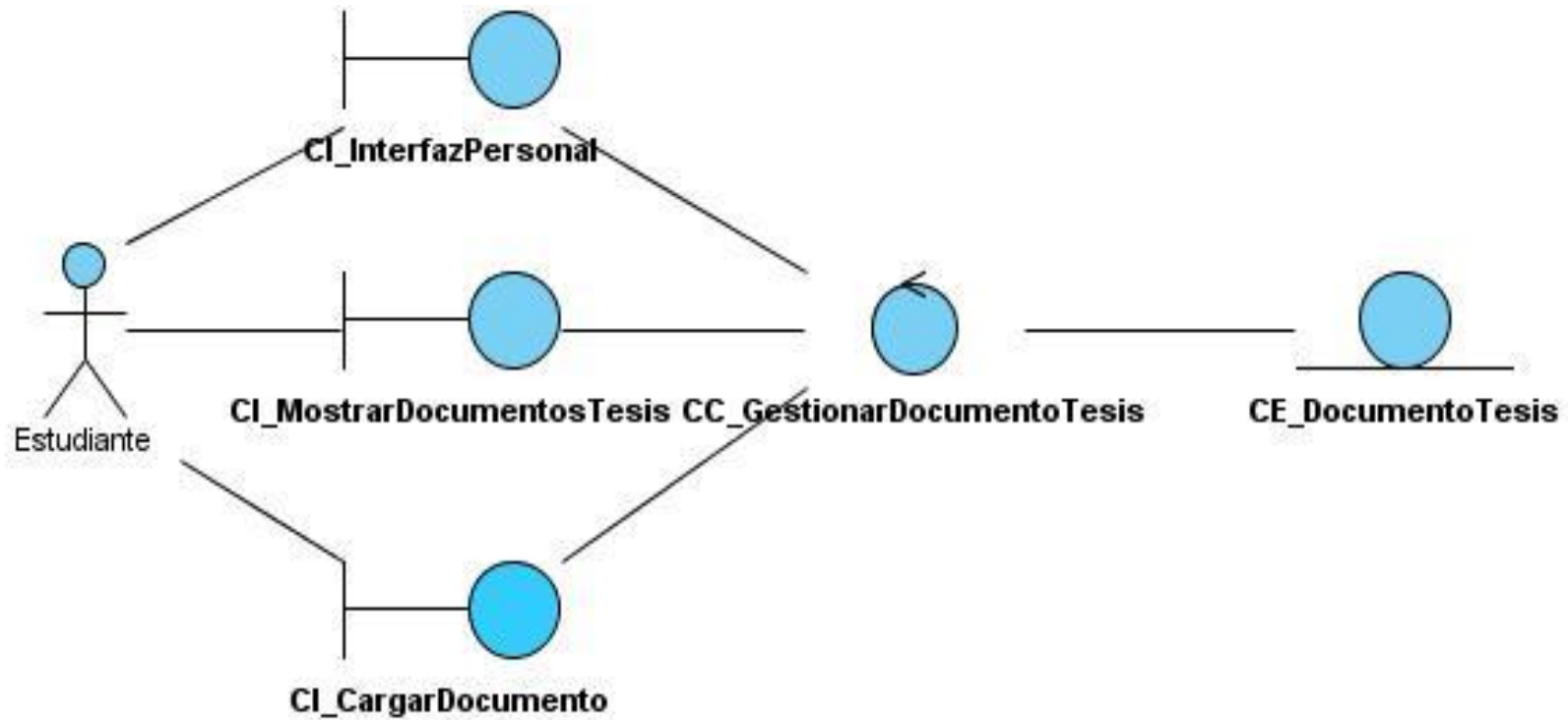


Figura 9: Diagrama de Clases del Análisis del CUS "Gestionar Documentos de Tesis"

Caso de Uso: "Gestionar Tribunal de Tesis"

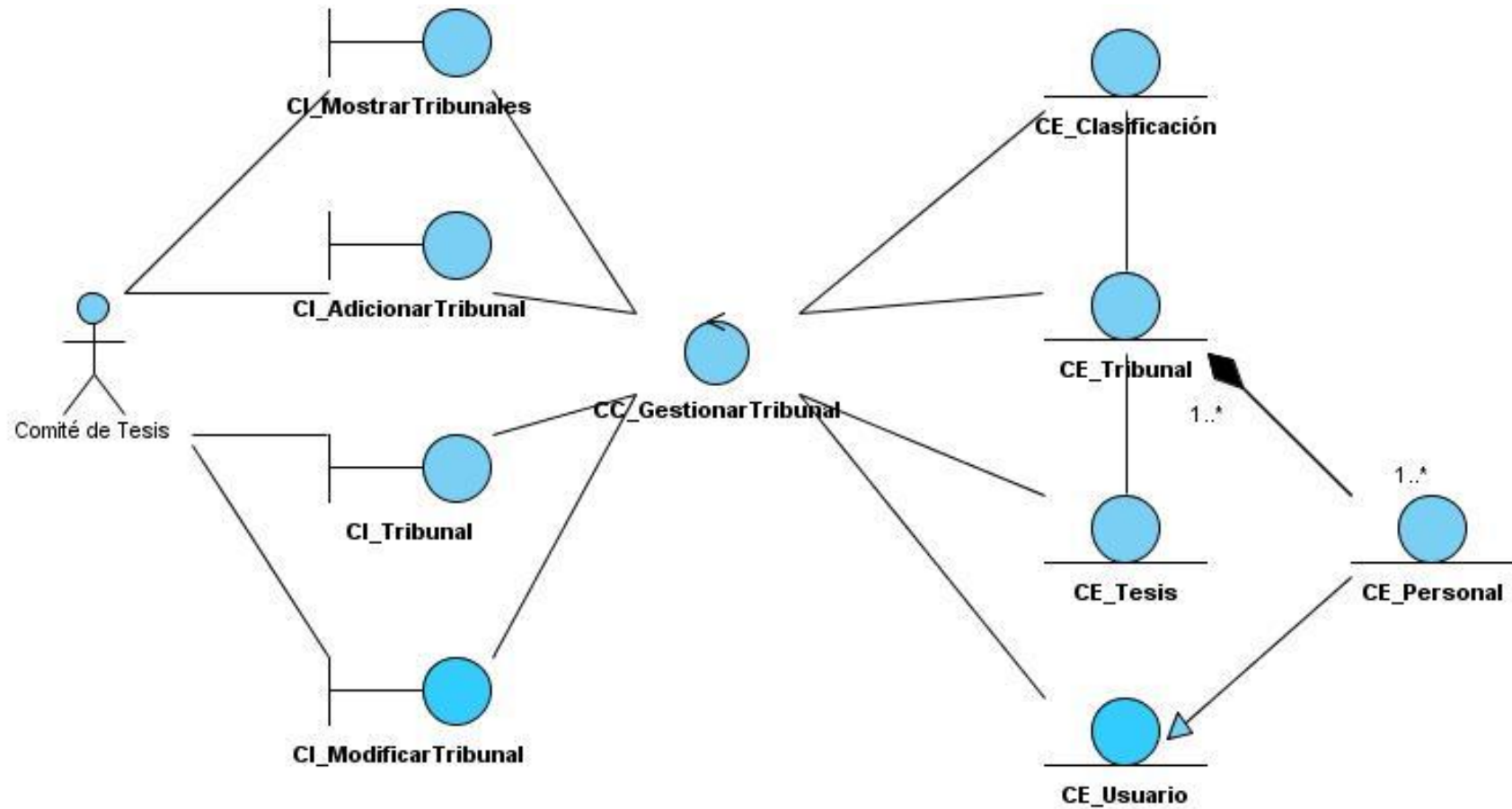


Figura 10: Diagrama de Clases del Análisis del CUS "Gestionar Tribunal de Tesis"

Caso de Uso: "Realizar Comentarios"

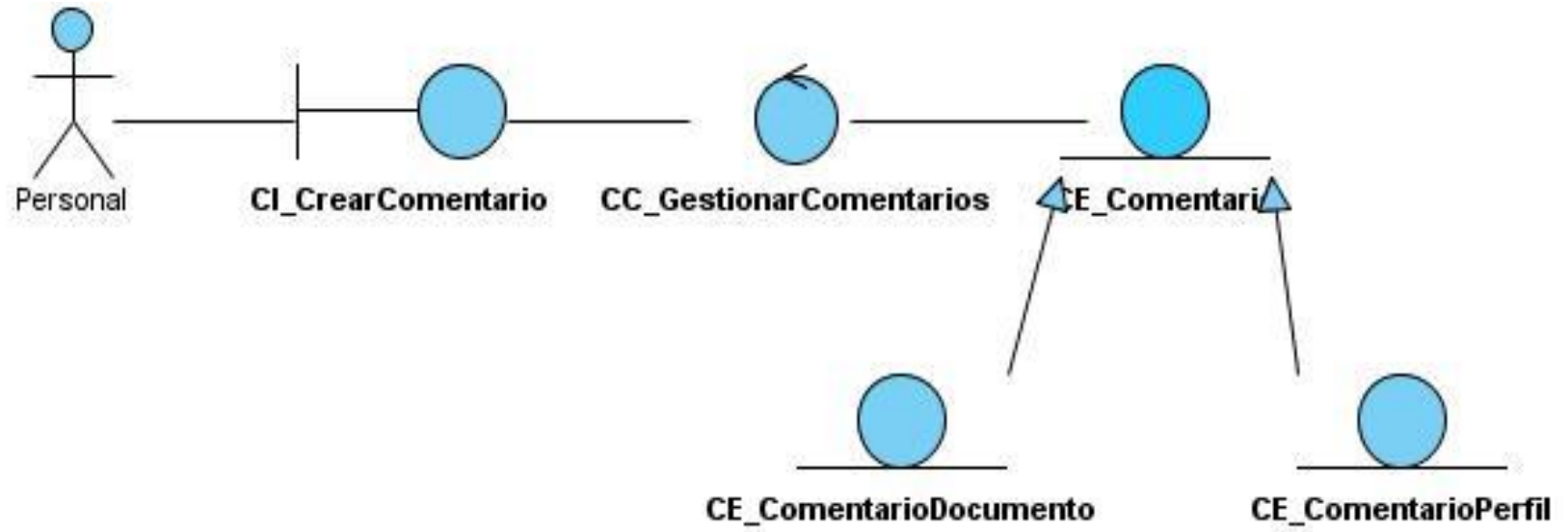


Figura 11: Diagrama de Clases del Análisis del CUS "Realizar Comentarios"

Caso de Uso: "Gestionar Proyectos"

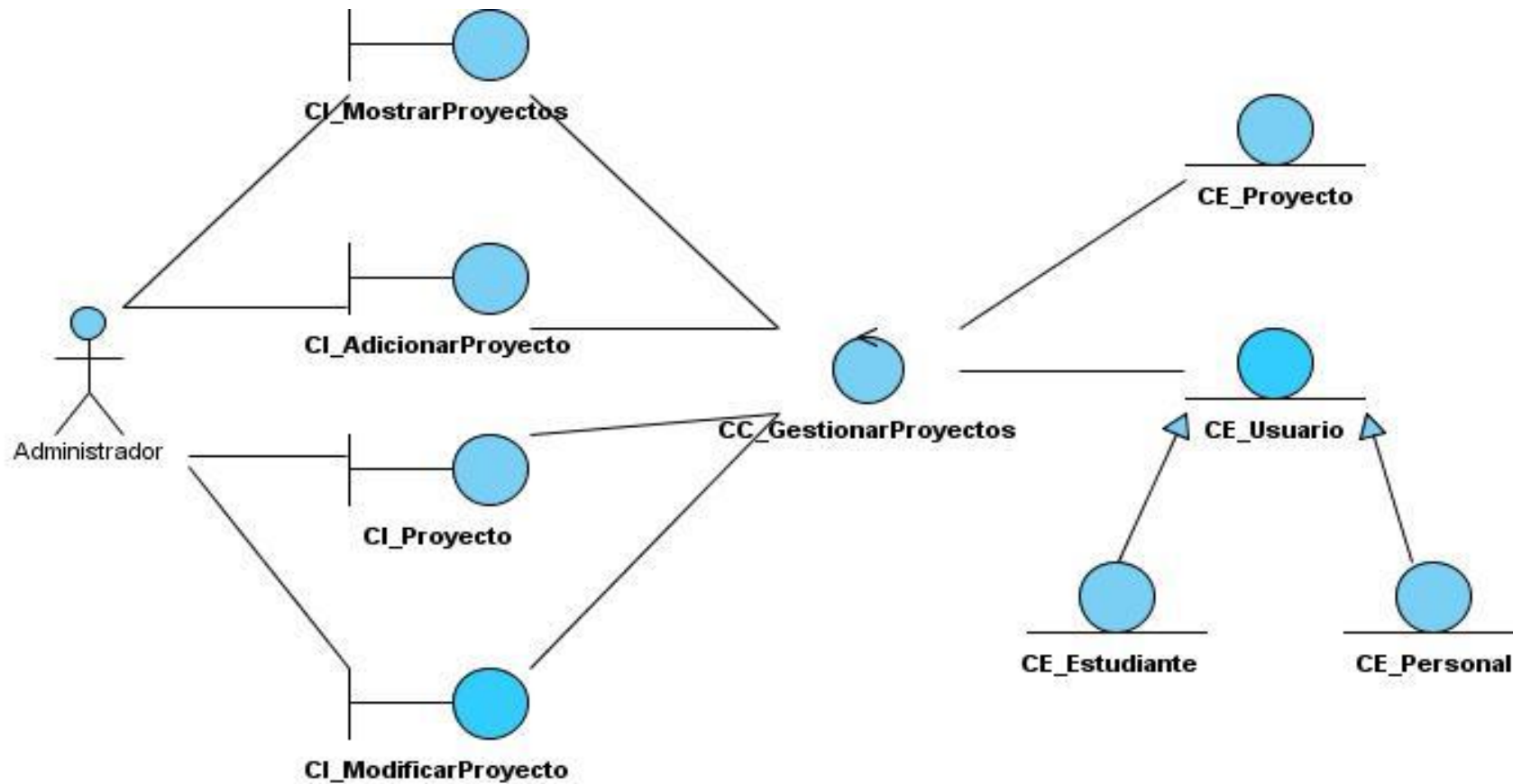


Figura 12: Diagrama de Clases del Análisis del CUS "Gestionar Proyectos"

Caso de Uso: “Gestionar Comentarios”

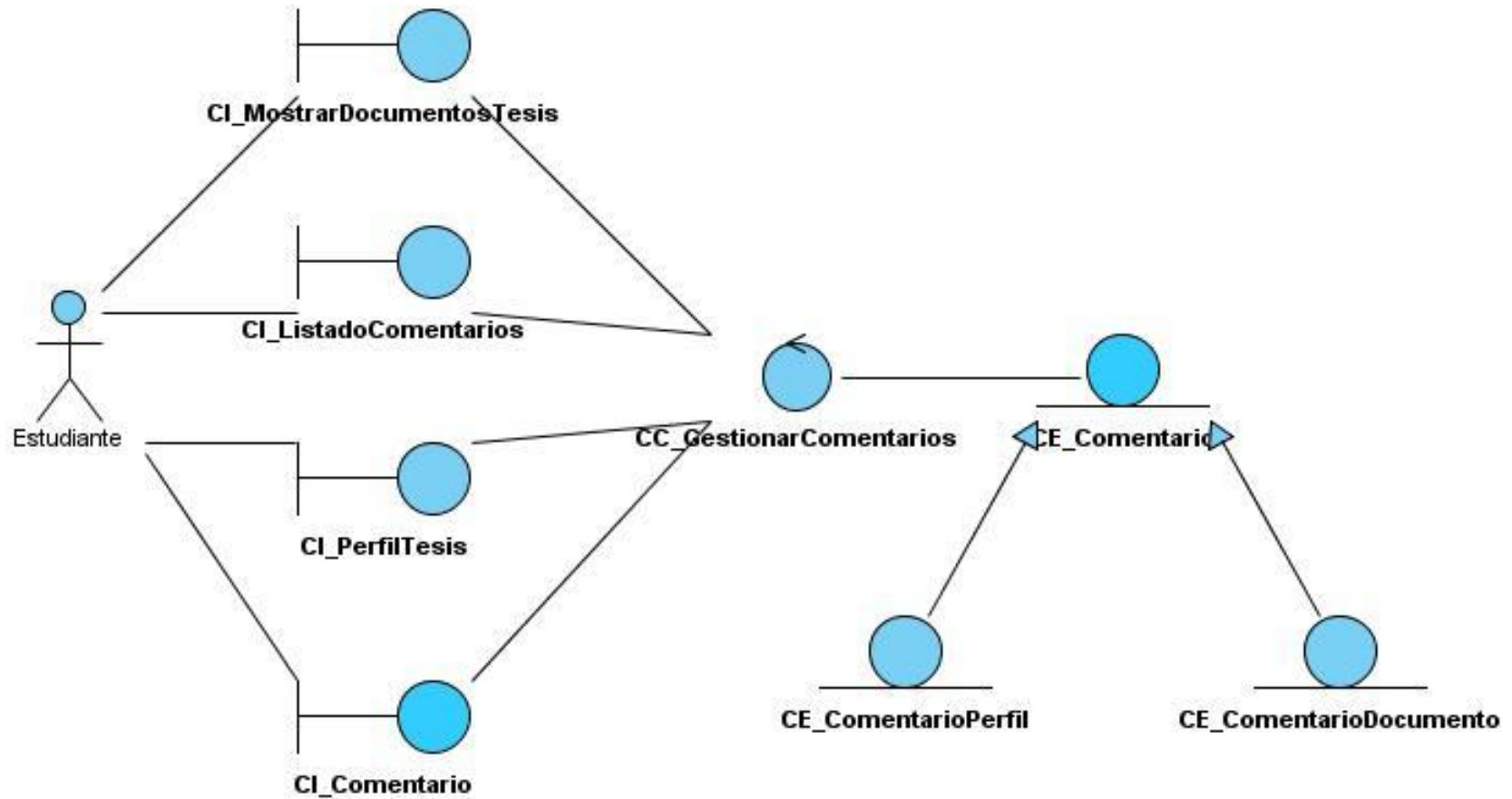


Figura 13: Diagrama de Clases del Análisis del CUS “Gestionar Comentarios”

Caso de Uso: "Gestionar Usuarios"

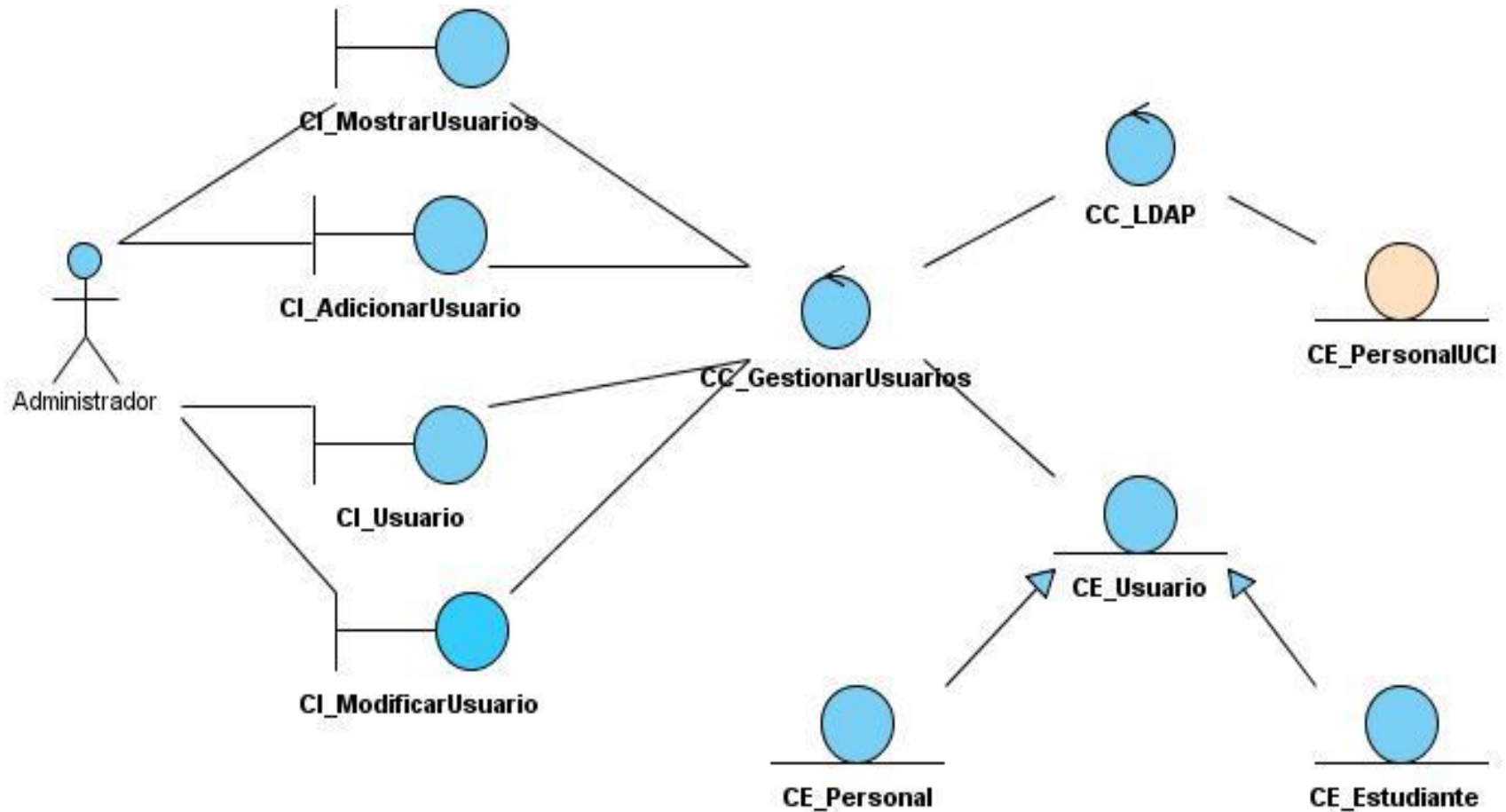


Figura 14: Diagrama de Clases del Análisis del CUS "Gestionar Usuarios"

Caso de Uso: "Gestionar Propuestas de Tesis"

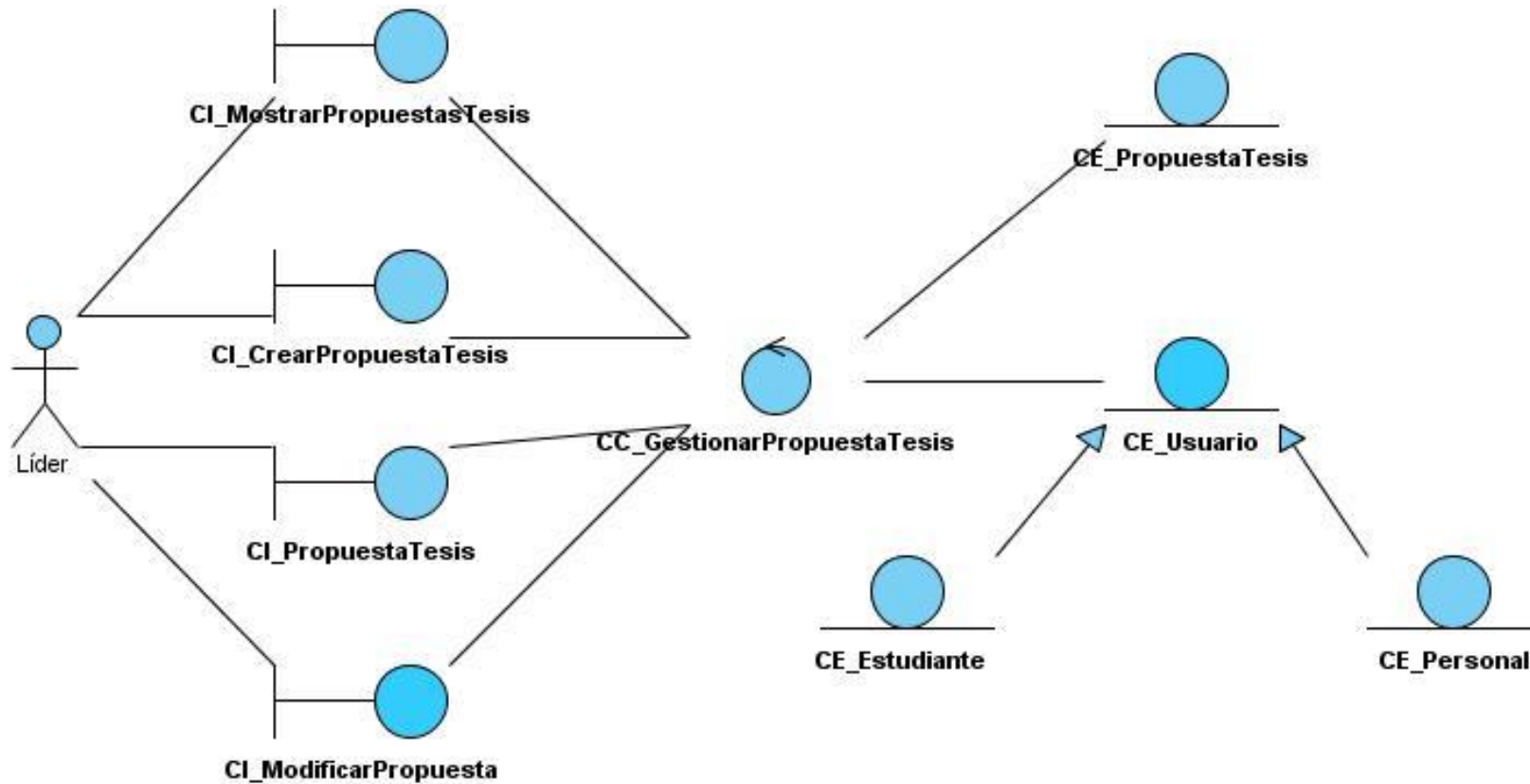


Figura 15: Diagrama de Clases del Análisis del CUS "Gestionar Propuestas de Tesis"

Caso de Uso: "Gestionar Evaluaciones"

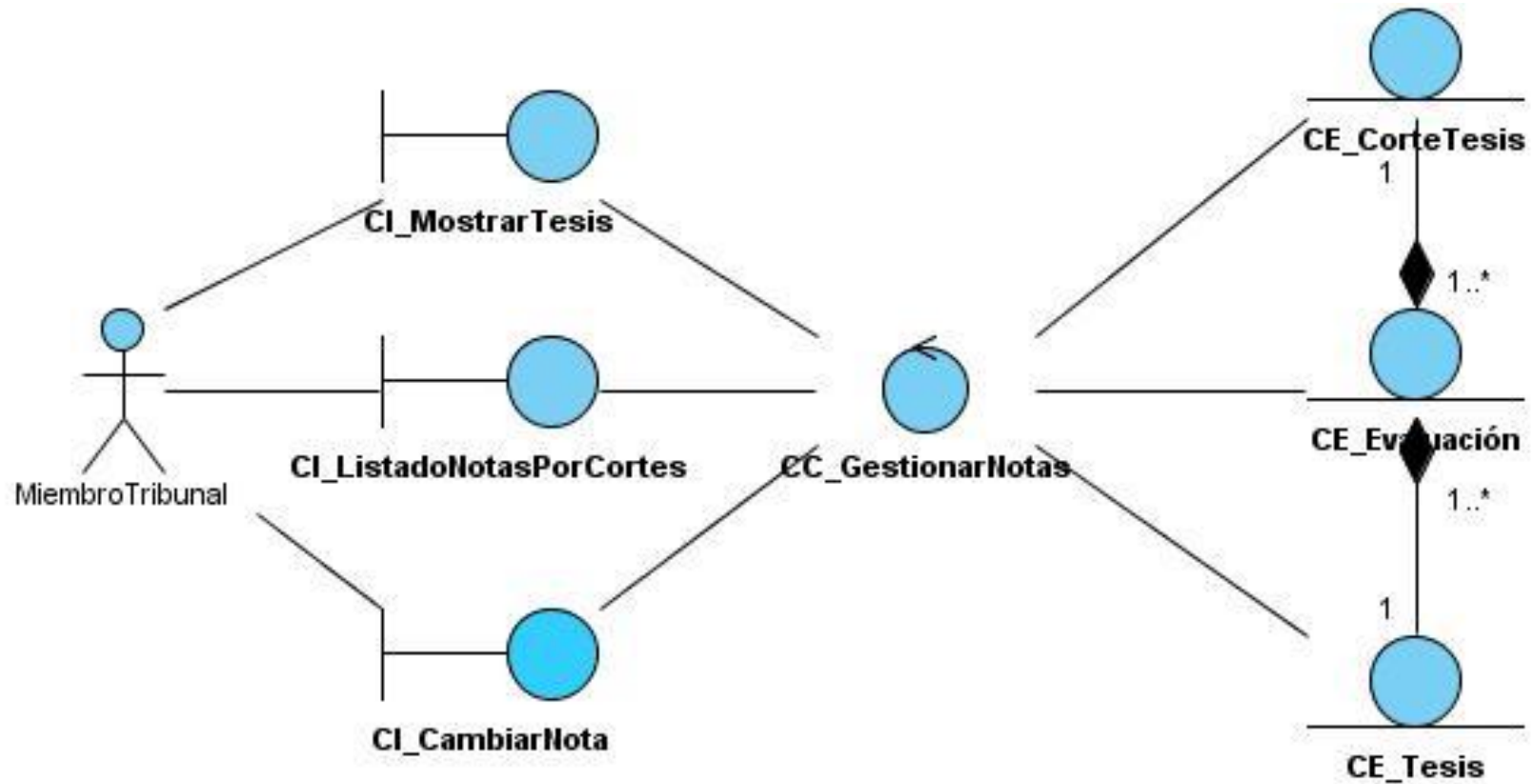


Figura 16: Diagrama de Clases del Análisis del CUS "Gestionar Evaluaciones"

Caso de Uso: "Realizar Búsquedas"

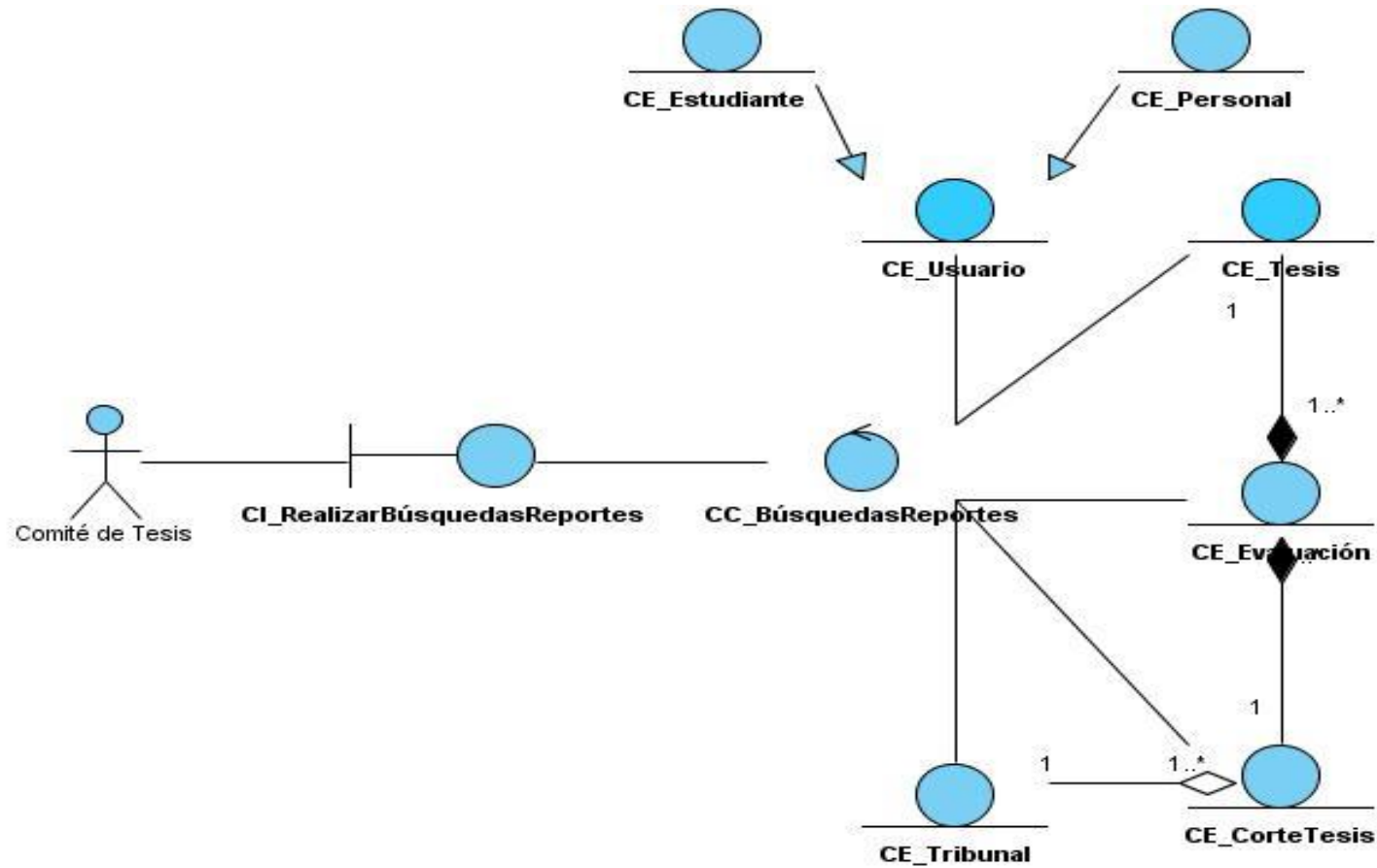


Figura 17: Diagrama de Clases del Análisis del CUS "Realizar Búsquedas"

3.4 Diseño del sistema

El diseño es un refinamiento del análisis que tiene en cuenta los requisitos no funcionales, o sea, CÓMO cumple el sistema sus objetivos. El mismo debe ser suficiente para que el sistema pueda ser implementado sin ambigüedades. El resultado final más importante de en esta etapa del desarrollo será el modelo de diseño.

3.4.1 Modelo del Diseño

El modelo de diseño es un modelo de objetos que describe la realización física de los casos de uso centrándose en cómo los requisitos funcionales y no funcionales, junto con otras restricciones relacionadas con el entorno de implementación, tienen impacto en el sistema a considerar. Sirve de abstracción de la implementación y es utilizado como entrada fundamental de las actividades de implementación.

3.4.2 Diagramas de clases del diseño

Los diagramas de clases se utilizan para modelar la vista de diseño estática de un sistema. Principalmente, esto incluye modelar el vocabulario del sistema, modelar las colaboraciones o modelar esquemas. Los diagramas de clases también son la base para un par de diagramas relacionados: los diagramas de componentes y los diagramas de despliegue.

A continuación se presentan los Diagramas de Clases del Diseño (DCD) por cada caso de uso del sistema (CUS):

Caso de Uso: "Autenticar Usuario"

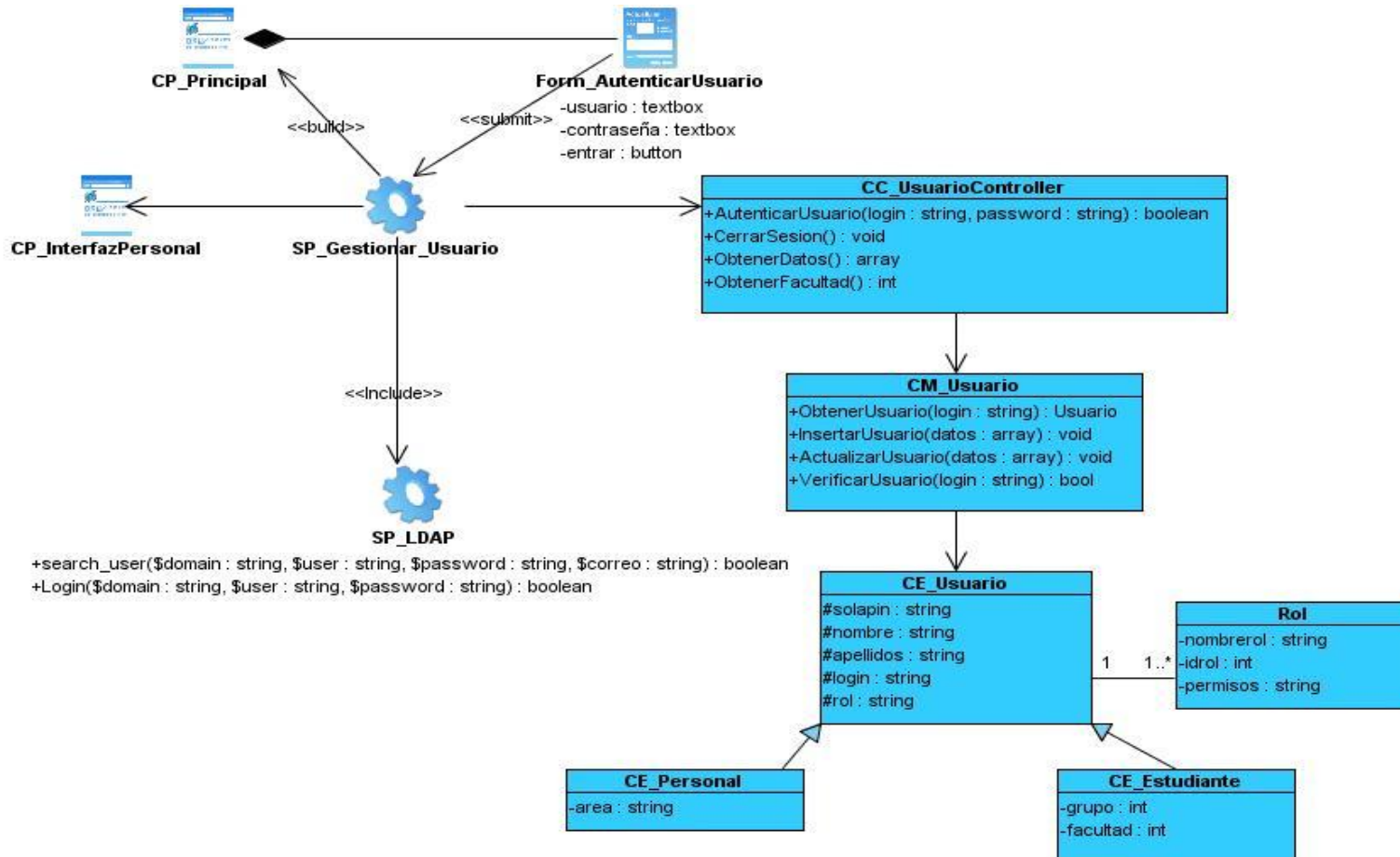


Figura 18: Diagrama de Clases del Diseño del CUS "Autenticar Usuario"

Caso de Uso: "Evaluar Perfil"

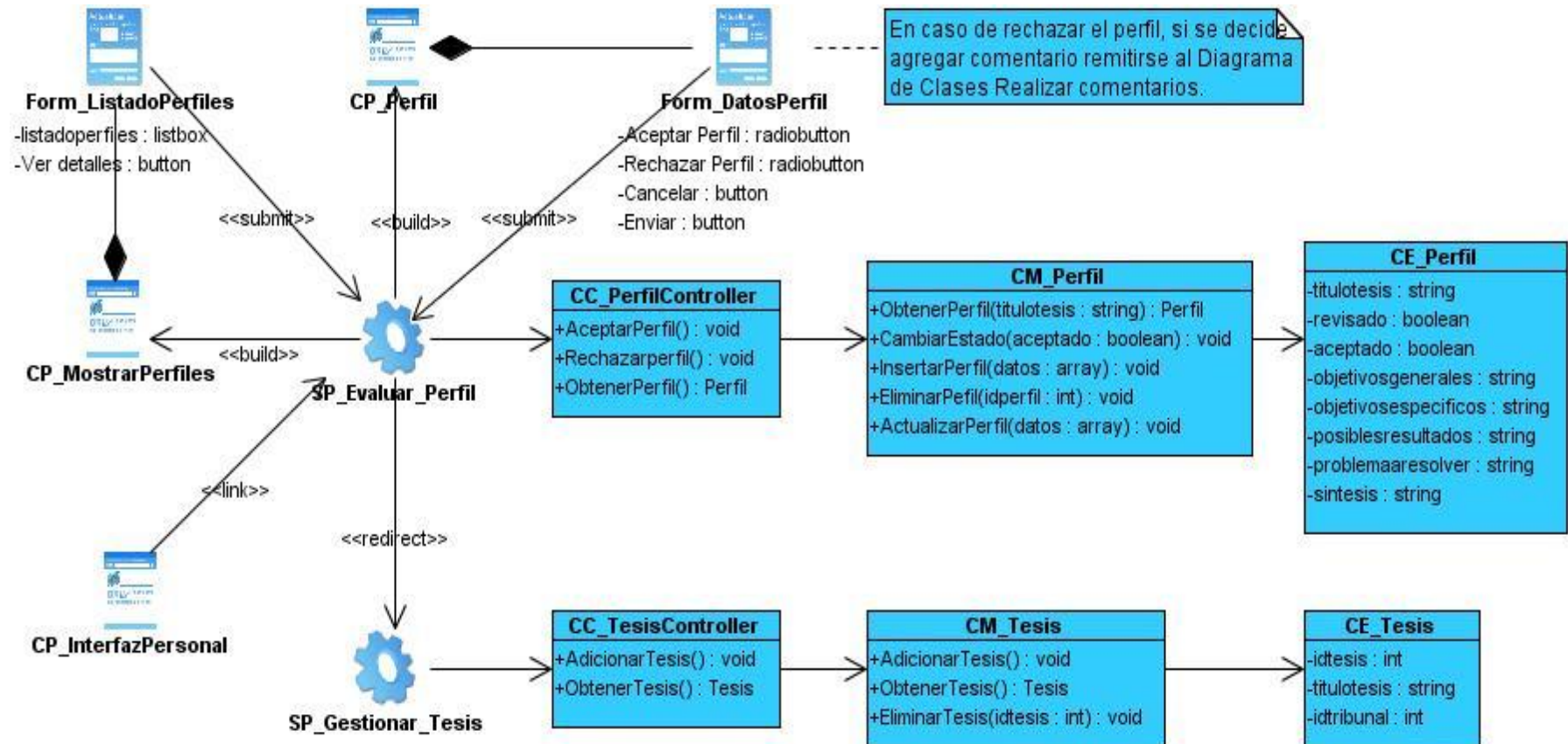


Figura 19: Diagrama de Clases del Diseño del CUS "Evaluar Perfil"

Caso de Uso: "Gestionar Perfil de Tesis"

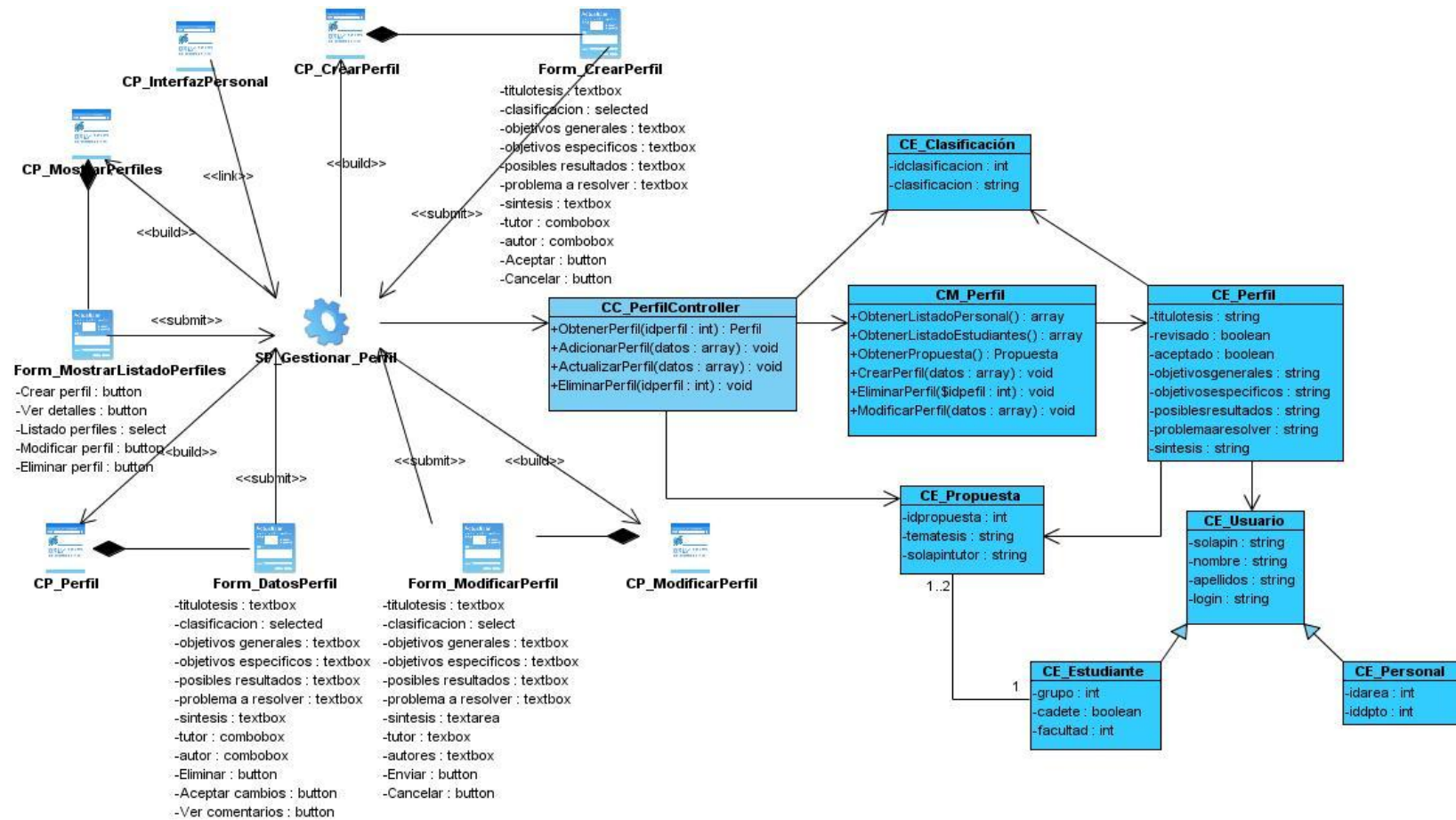


Figura 20: Diagrama de Clases del Diseño del CUS "Gestionar Perfil de Tesis"

Caso de Uso: “Gestionar Cortes de Tesis”

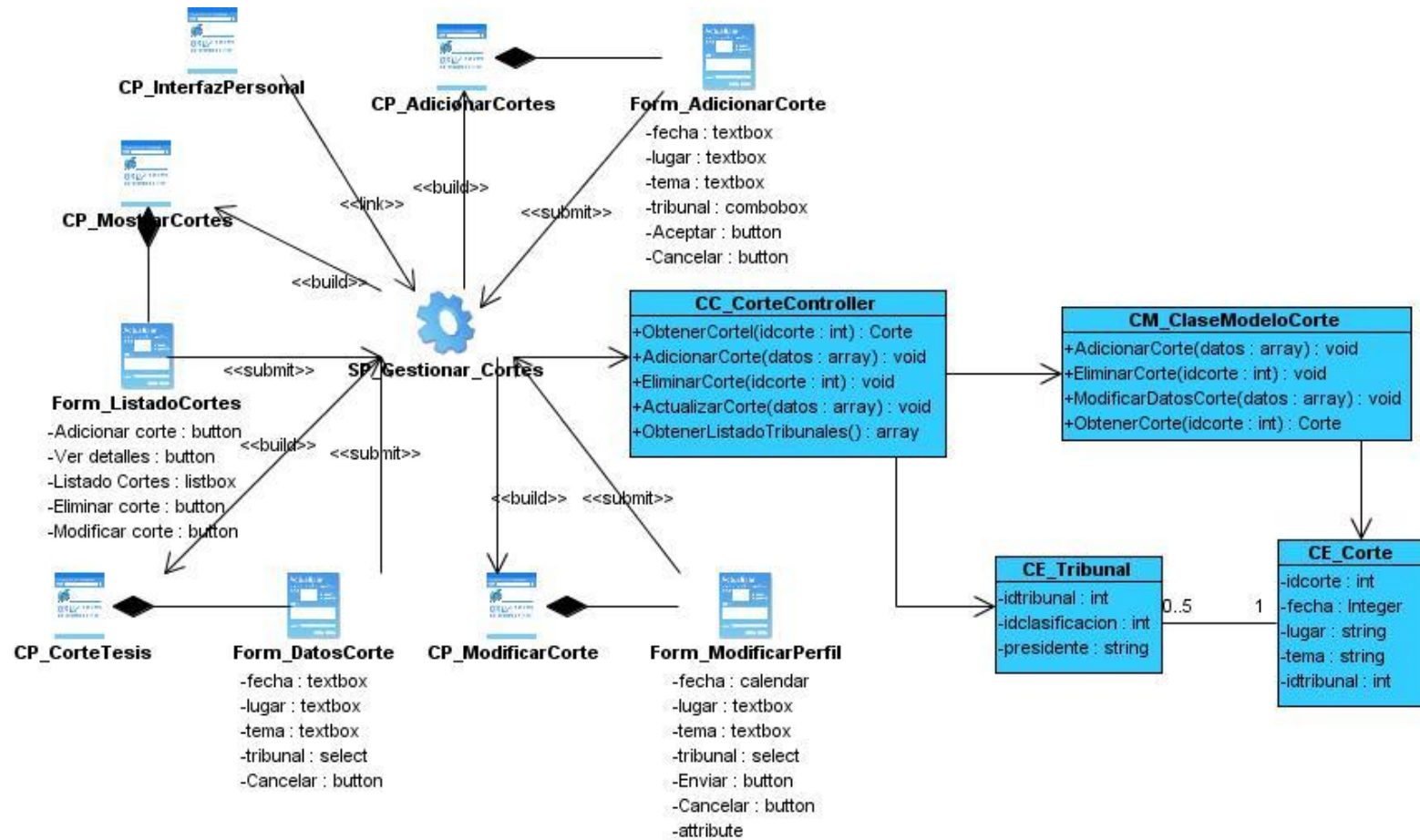


Figura 21: Diagrama de Clases del Diseño del CUS “Gestionar Cortes de Tesis”

Caso de Uso: “Gestionar Documentos de Tesis”

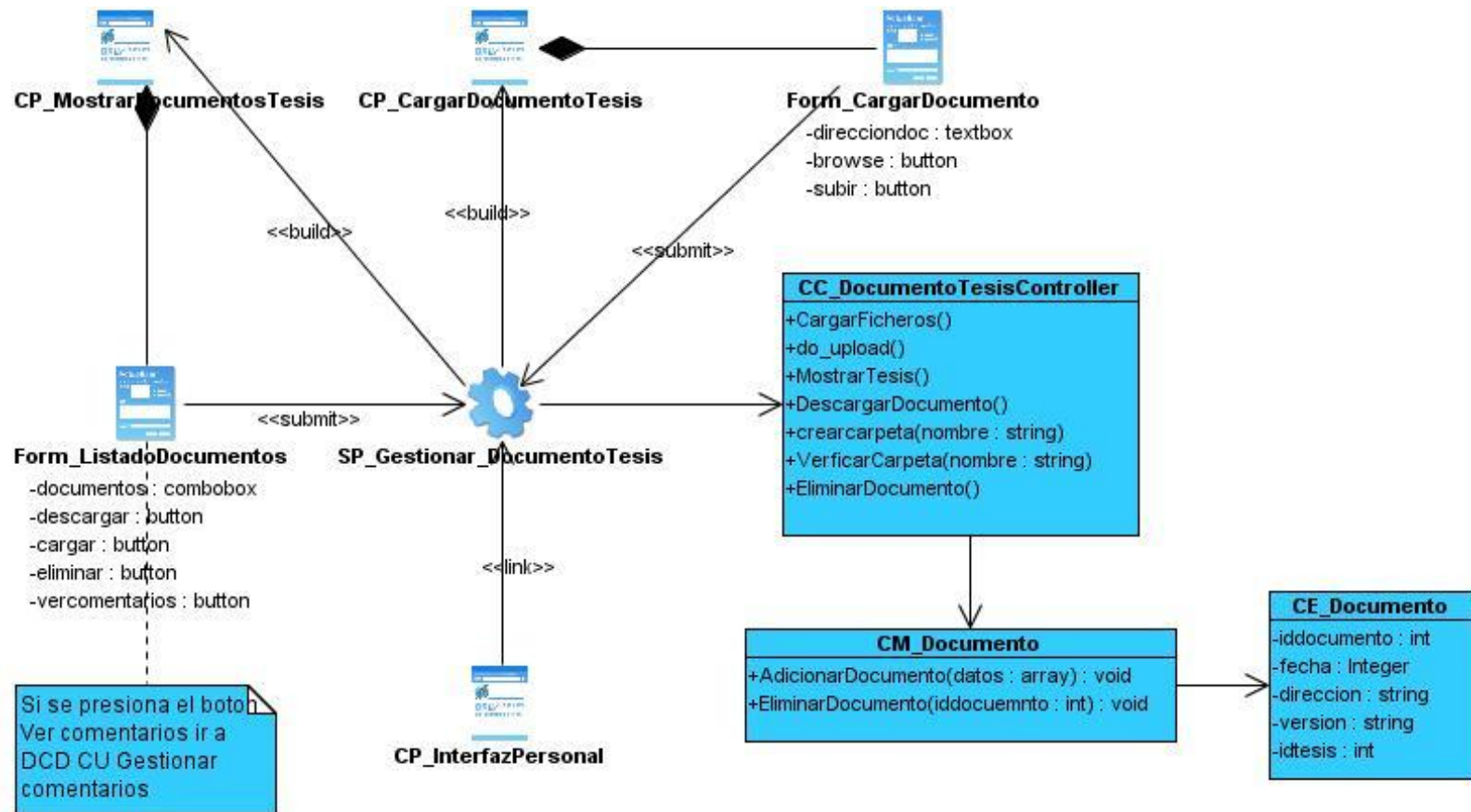


Figura 22: Diagrama de Clases del Diseño del CUS “Gestionar Documentos de Tesis”

Caso de Uso: “Gestionar Tribunal de Tesis”

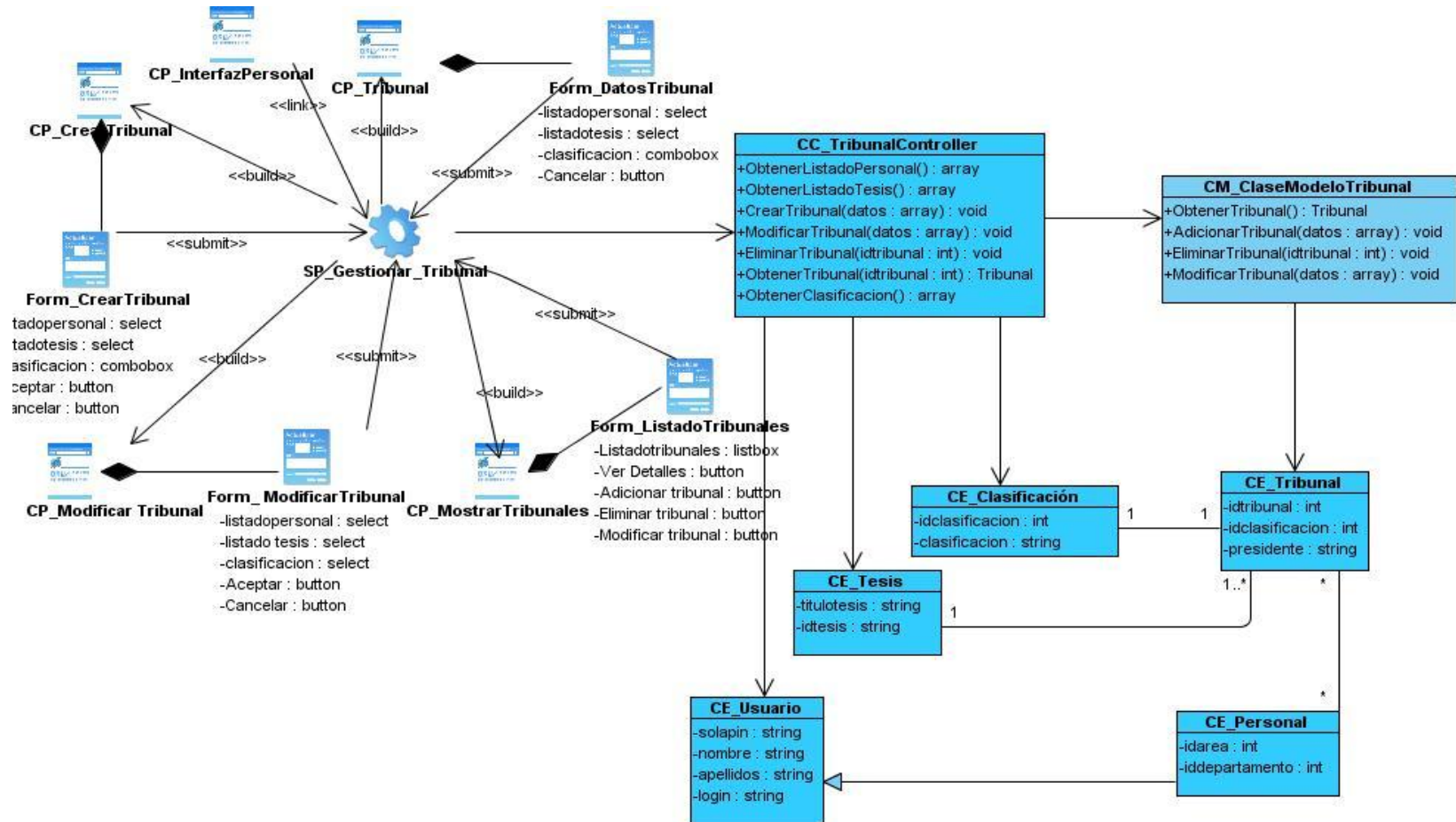


Figura 23: Diagrama de Clases del Diseño del CUS “Gestionar Tribunal de Tesis”

Caso de Uso: "Gestionar Proyectos"

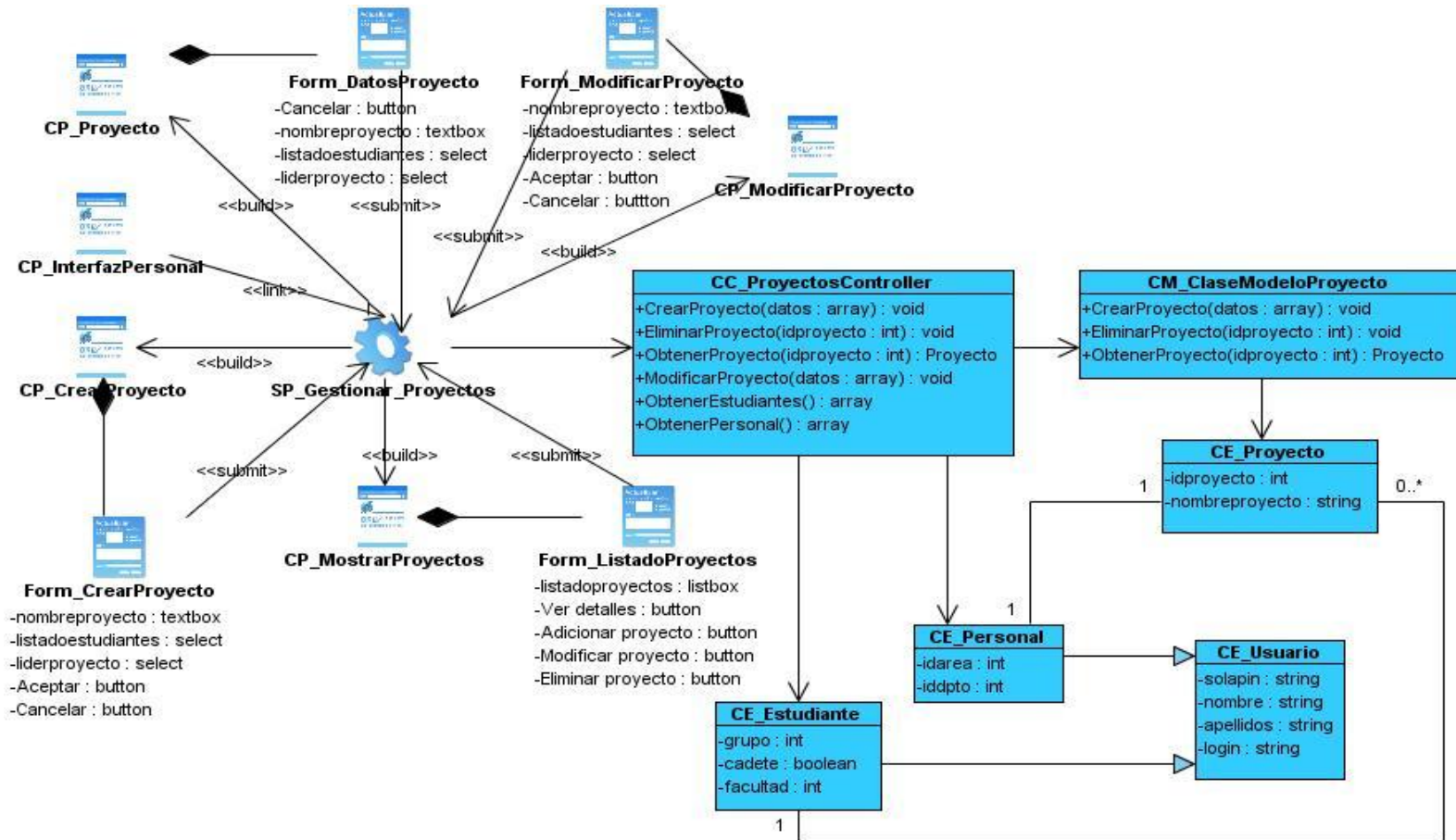


Figura 24: Diagrama de Clases del Diseño del CUS "Gestionar Proyectos"

Caso de Uso: "Realizar Comentarios"

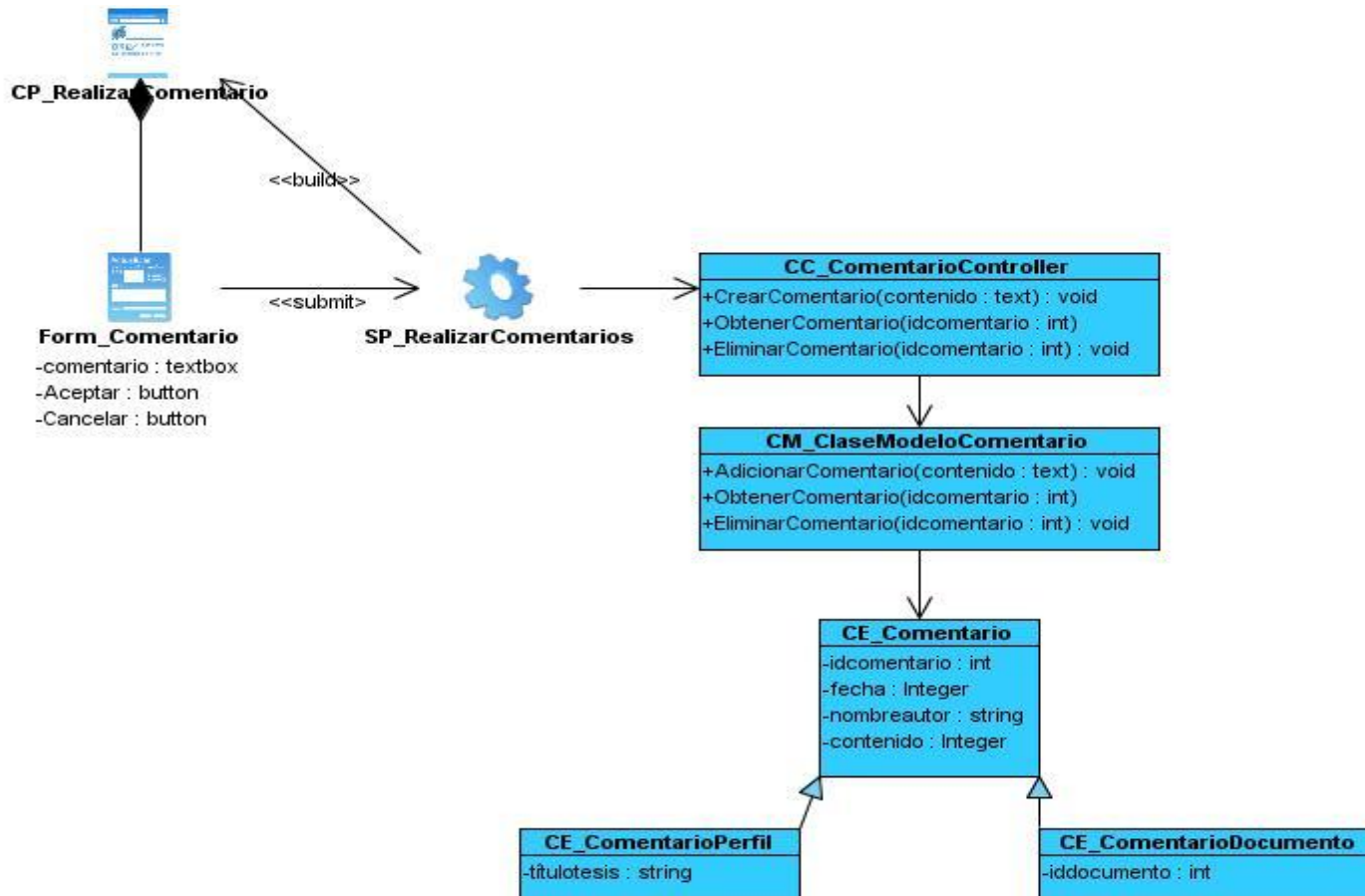


Figura 25: Diagrama de Clases del Diseño del CUS "Realizar Comentarios"

Caso de Uso: “Gestionar Comentarios”

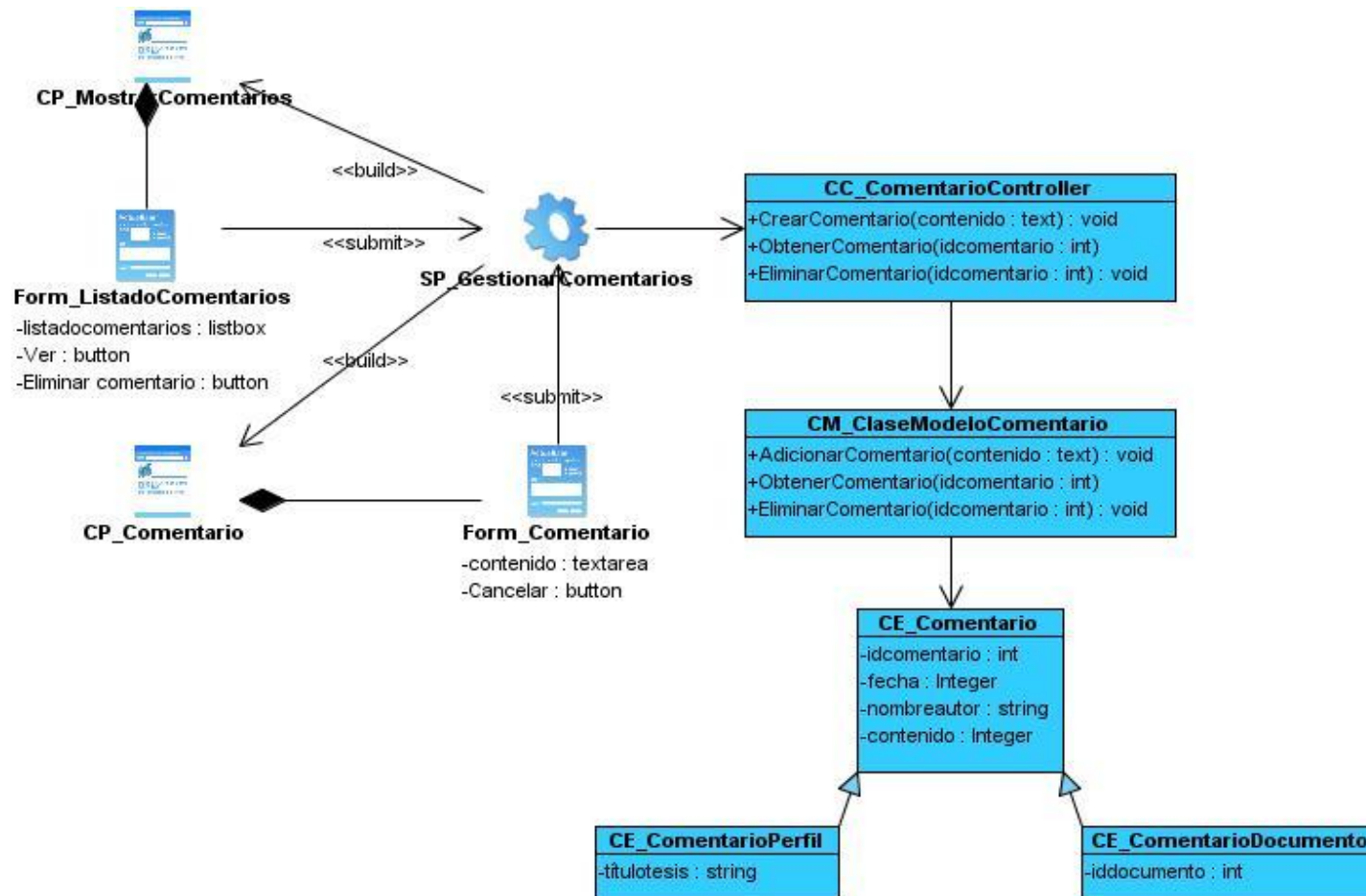


Figura 26: Diagrama de Clases del Diseño del CUS “Gestionar Comentarios”

Caso de Uso: “Gestionar Usuarios”

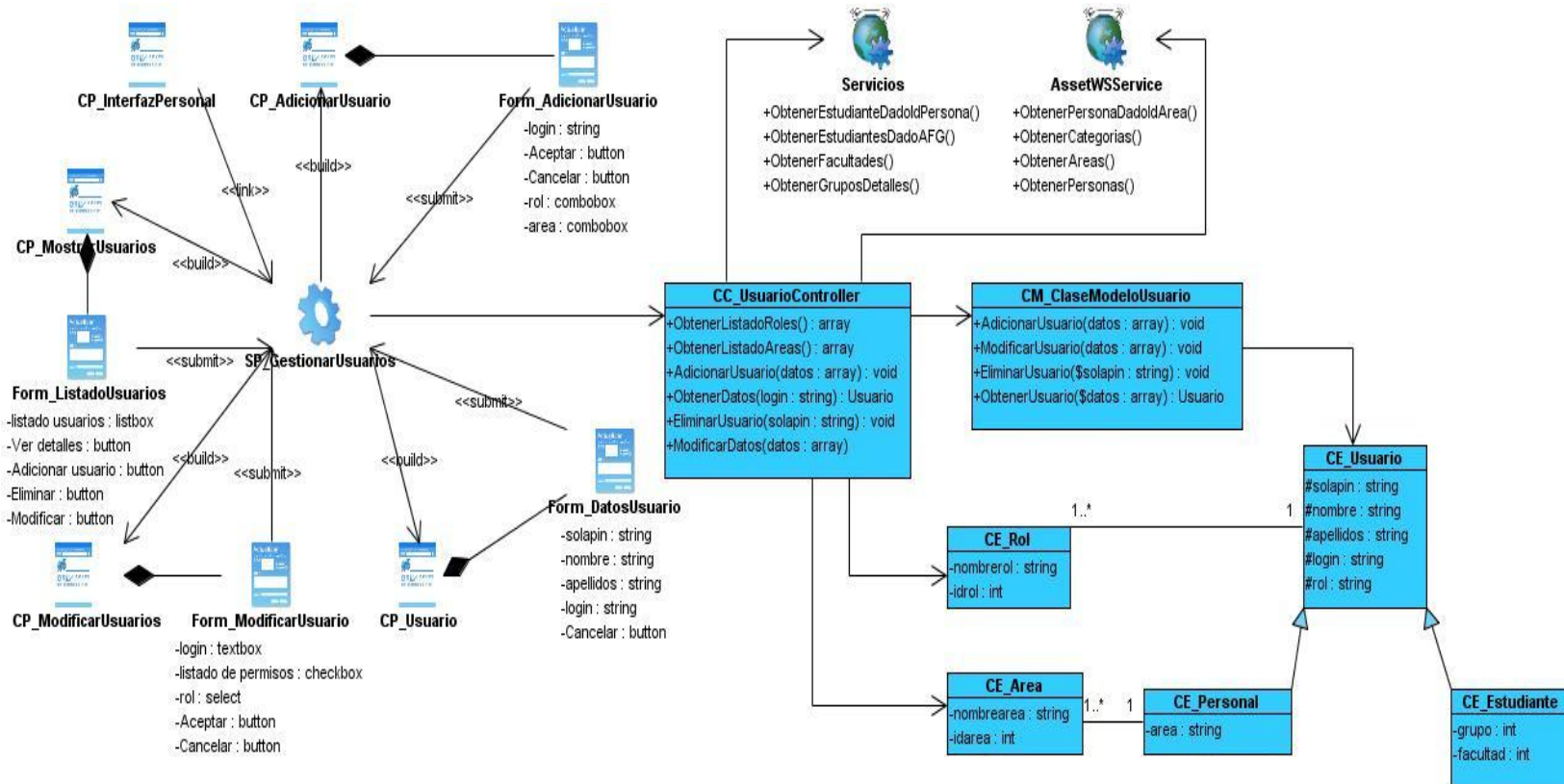


Figura 27: Diagrama de Clases del Diseño del CUS “Gestionar Usuarios”

Caso de Uso: “Gestionar Propuestas de Tesis”

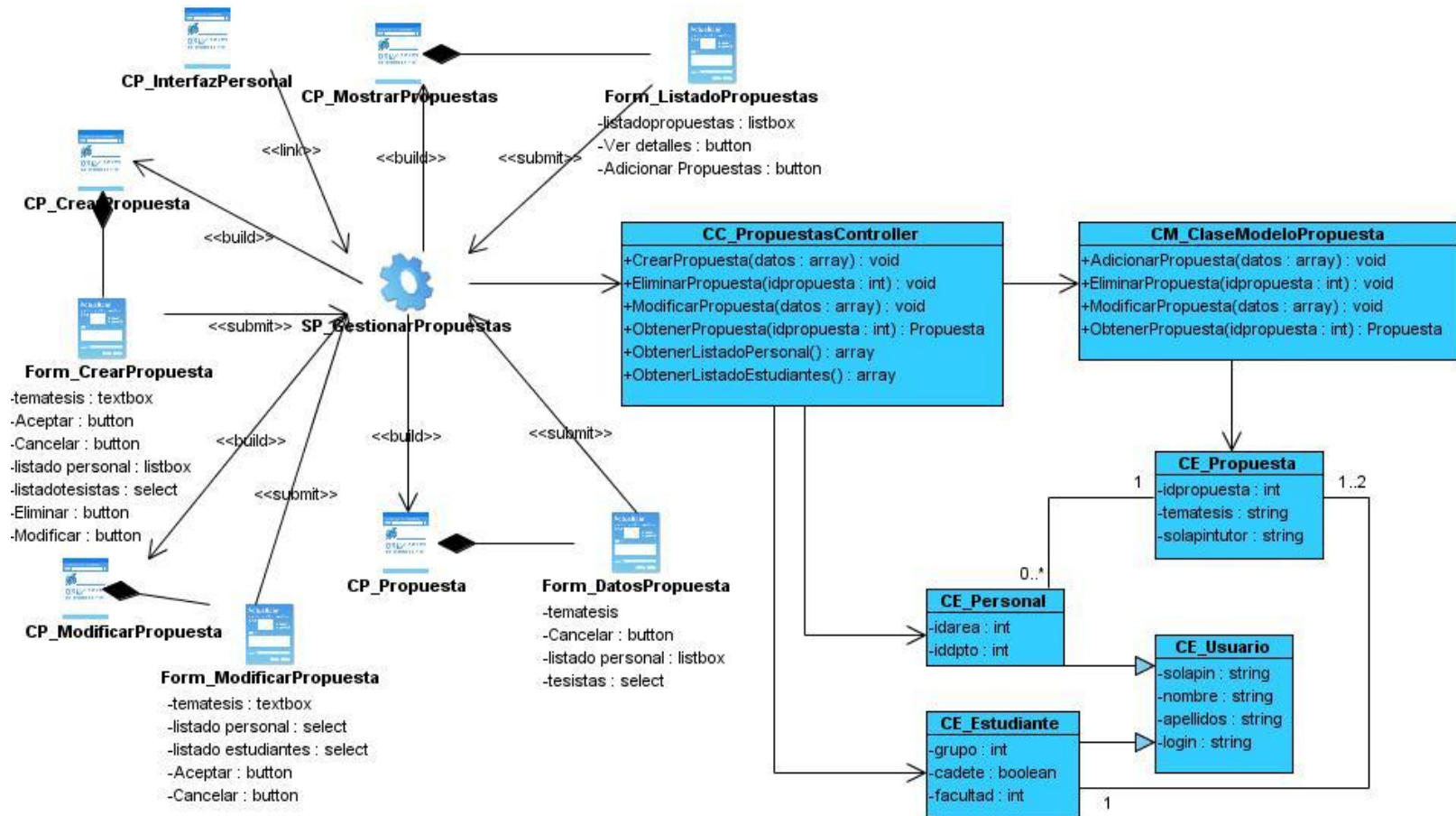


Figura 28: Diagrama de Clases del Diseño del CUS “Gestionar Propuestas”

Caso de Uso: “Gestionar Evaluaciones”

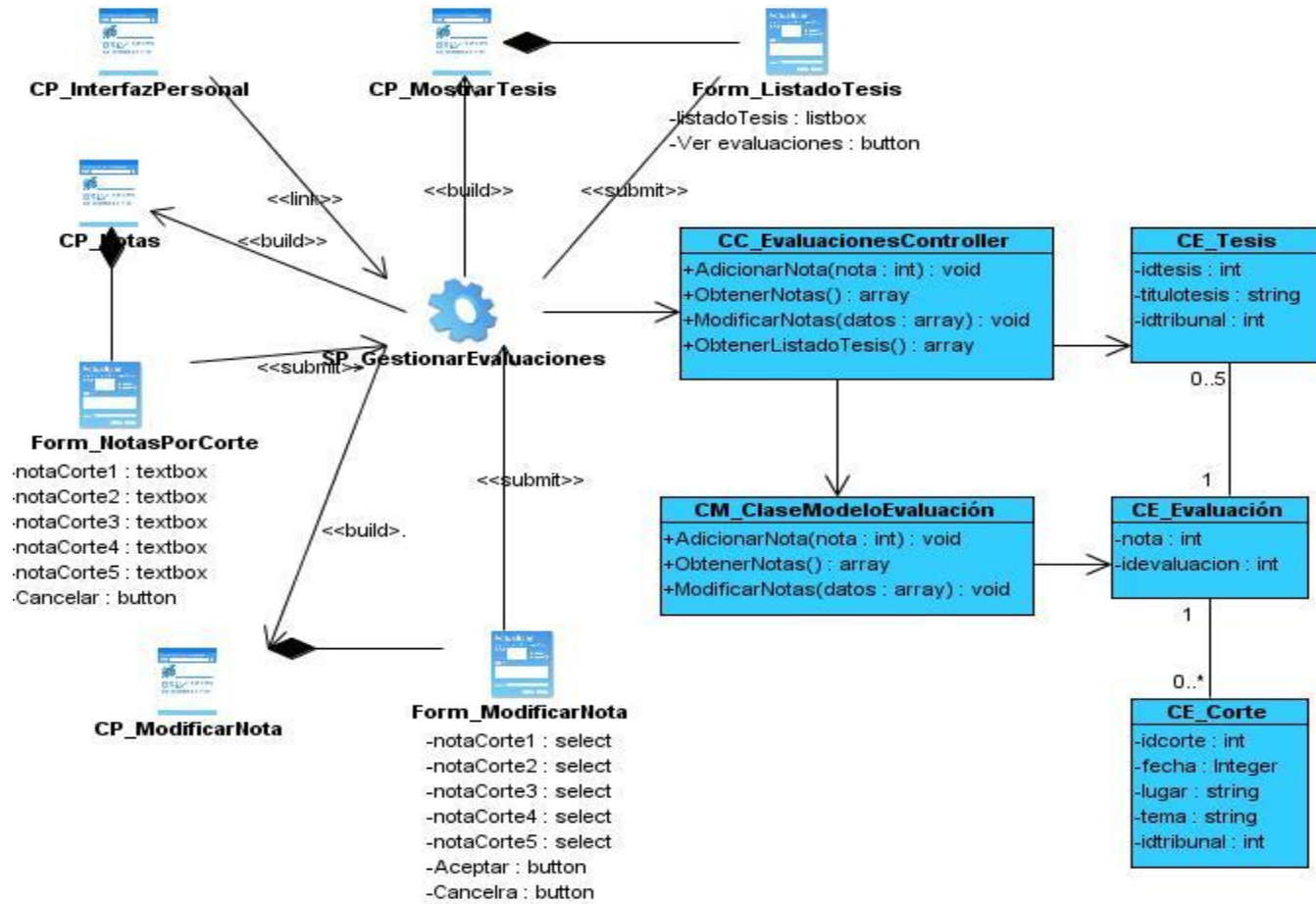


Figura 29: Diagrama de Clases del Diseño del CUS “Gestionar Evaluaciones”

Caso de Uso: "Realizar Reportes"

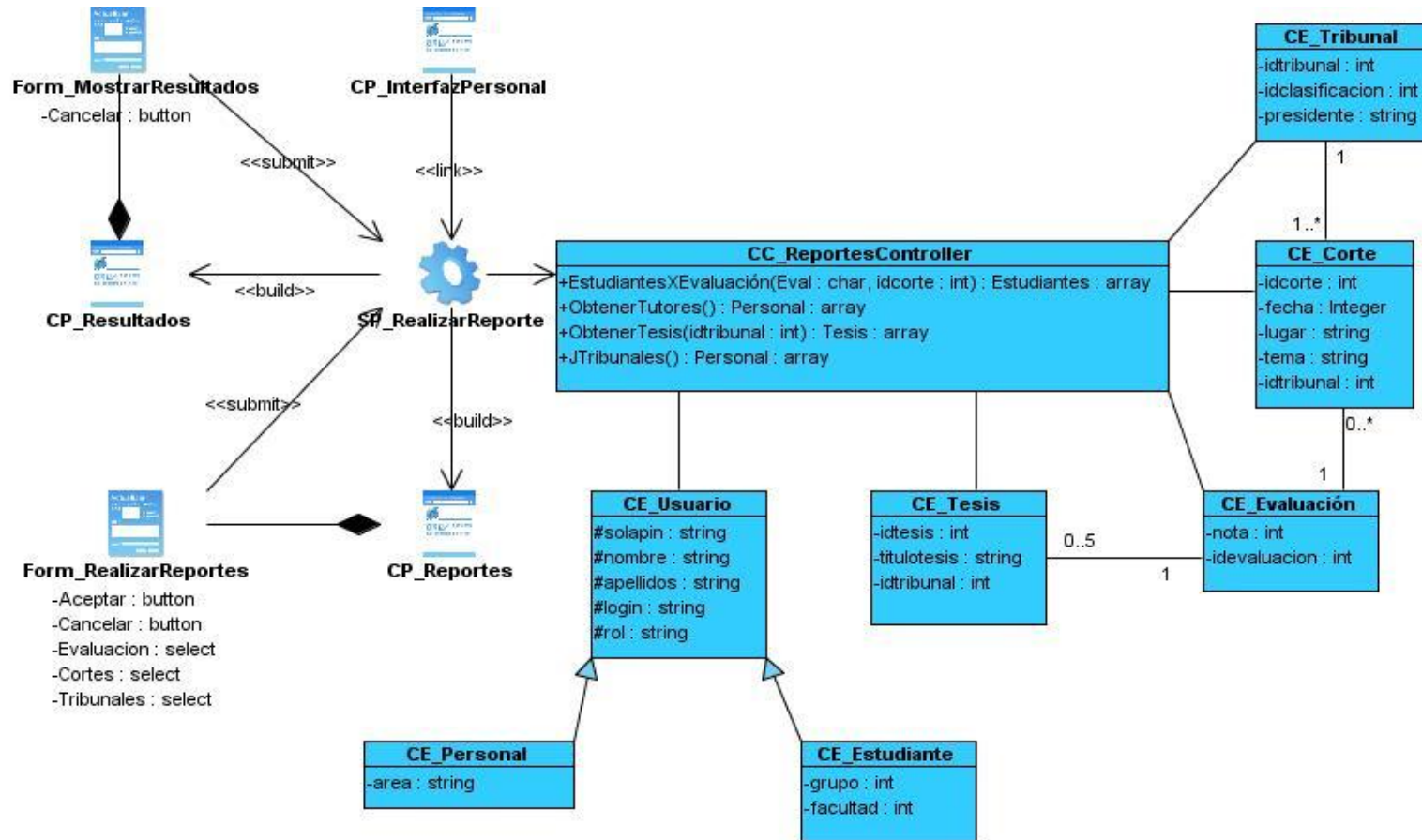


Figura 30: Diagrama de Clases del Diseño del CUS "Realizar Reportes"

3.4.3 Patrones de diseño

Los principios del diseño que se requieren para construir buenos diagramas de interacción y de forma general el modelo de diseño, pueden codificarse, explicarse y utilizarse en forma metódica. Esta manera de entender y usar los principios del diseño se funda en los patrones con que se asignan las responsabilidades [25]

Los patrones GRASP describen los principios fundamentales de la asignación de responsabilidades a objetos. GRASP es un acrónimo que significa General Responsibility Assignment Software Patterns (Patrones Generales de Software para Asignar Responsabilidades) [25]. A continuación se describen los patrones aplicados:

Para la lograr una asignación eficiente de las responsabilidades, a las clases utilizadas, se aplicó el patrón Experto, el cual consiste en asignar una responsabilidad al experto en información, o sea, a la clase que cuenta con la información necesaria para cumplir la responsabilidad. Este es un patrón que se usa más que cualquier otro al asignar responsabilidades, ya que es un principio básico que suele utilizarse en el diseño orientado a objetos. El mismo se evidencia, por ejemplo, en el diagrama de clases del caso de uso del sistema Gestionar Roles, donde la clase Modelo Rol es la encargada de realizar las consultas a la base de datos, de forma que Roles Controller no tenga que establecer una relación directa con la entidad Rol.

Además, el framework que se utiliza para desarrollar la presente aplicación, CodeIgniter, fue creado desde un punto de vista técnico y arquitectónico, entre otros aspectos, con la idea de garantizar un poco acoplamiento, o sea, mientras menos componentes dependan de otro, más reusable y flexible se torna el sistema, ya que las responsabilidades están distribuidas entre las clases, de manera que su colocación no incrementa el acoplamiento¹⁵, pero a su vez se mantienen enfocadas y relacionadas las responsabilidades de las mismas, lo que responde al patrón de Alta Cohesión¹⁶.

¹⁵ Acoplamiento, medida de la fuerza con que una clase está conectada a otras.

¹⁶ Cohesión, medida de cuán relacionadas y enfocadas están las responsabilidades de una clase.

Con la aplicación de los patrones descritos anteriormente se logró realizar la asignación correcta de las responsabilidades durante el diseño de la aplicación.

3.4.4 Diagramas de interacción

Los diagramas de interacción se utilizan para modelar los aspectos dinámicos de un sistema, lo que conlleva a modelar instancias concretas o prototípicas de clases interfaces, componentes y nodos, junto con los mensajes enviados entre ellos, todo en el contexto de un escenario que ilustra un comportamiento. [26] Estos diagramas se expresan de dos formas, una en diagramas de colaboración, que son los que muestran las relaciones entre los objetos y los mensajes que intercambian y, otra en diagramas de secuencia, que muestran las interacciones expresadas en función de secuencias temporales y fue a su vez el escogido para describir cada escenario de los casos de uso de la presente aplicación.

3.4.4.1 Diagramas de Secuencia

Un diagrama de secuencia muestra las interacciones entre objetos ordenadas en secuencia temporal. Muestra los objetos que se encuentran en el escenario y la secuencia de mensajes intercambiados entre los objetos para llevar a cabo la funcionalidad descrita por el escenario. Documentan el diseño desde el punto de vista de los casos de uso, observando qué mensajes se envían a los objetos, componentes o casos de uso y dan una aproximación del tiempo que consume el método invocado.

Los diagramas de secuencia de los flujos principales de los CU más significativos del sistema se muestran en los anexos. [VER ANEXO 5](#)

3.5 Diseño de la Base de Datos

Las Bases de datos necesitan de una definición de su estructura que le permitan almacenar datos, reconocer el contenido, y recuperar la información. Es por ello que uno de los pasos cruciales en la construcción de una aplicación que maneje una base de datos, es sin duda, el diseño de la misma. Esto permite que las tablas sean definidas apropiadamente y se garantice que tenga eficiencia. Al diseñar una base de datos, se refleja la estructura del problema en el mundo real, evita el almacenamiento de información redundante y permite además representar todos los datos esperados, incluso con el paso del tiempo.

Para lograr diseñar la Base de Datos se partió de la definición de las clases persistentes. La persistencia es la capacidad de un objeto de mantener su valor en el espacio y en el tiempo. Luego del refinamiento y clasificación de estas clases y sus atributos se realizó la construcción de un diagrama de clases persistentes.

3.5.1 Modelo lógico de datos. Diagrama de clases persistentes

El modelo lógico de datos representa la información que maneja el sistema, siendo una fuente de información para el modelo físico. A continuación se muestra el Modelo lógico de datos diseñado para la base de datos de la presente aplicación, el cual no es más que el diagrama de clases persistentes. El diagrama de clases persistentes es un diagrama de clases en el que, como bien indica su nombre, solo aparecen las clases persistentes, de las que hay que expandir detalles estructurales. Estas clases tuvieron como origen a las clasificadas como entidad, ya que estas modelan la información y el comportamiento asociado al fenómeno en cuestión.

3.5.2 Modelo físico de datos. Modelo de datos

A partir del diagrama de clases persistentes se obtuvo el modelo de datos que se muestra a continuación.

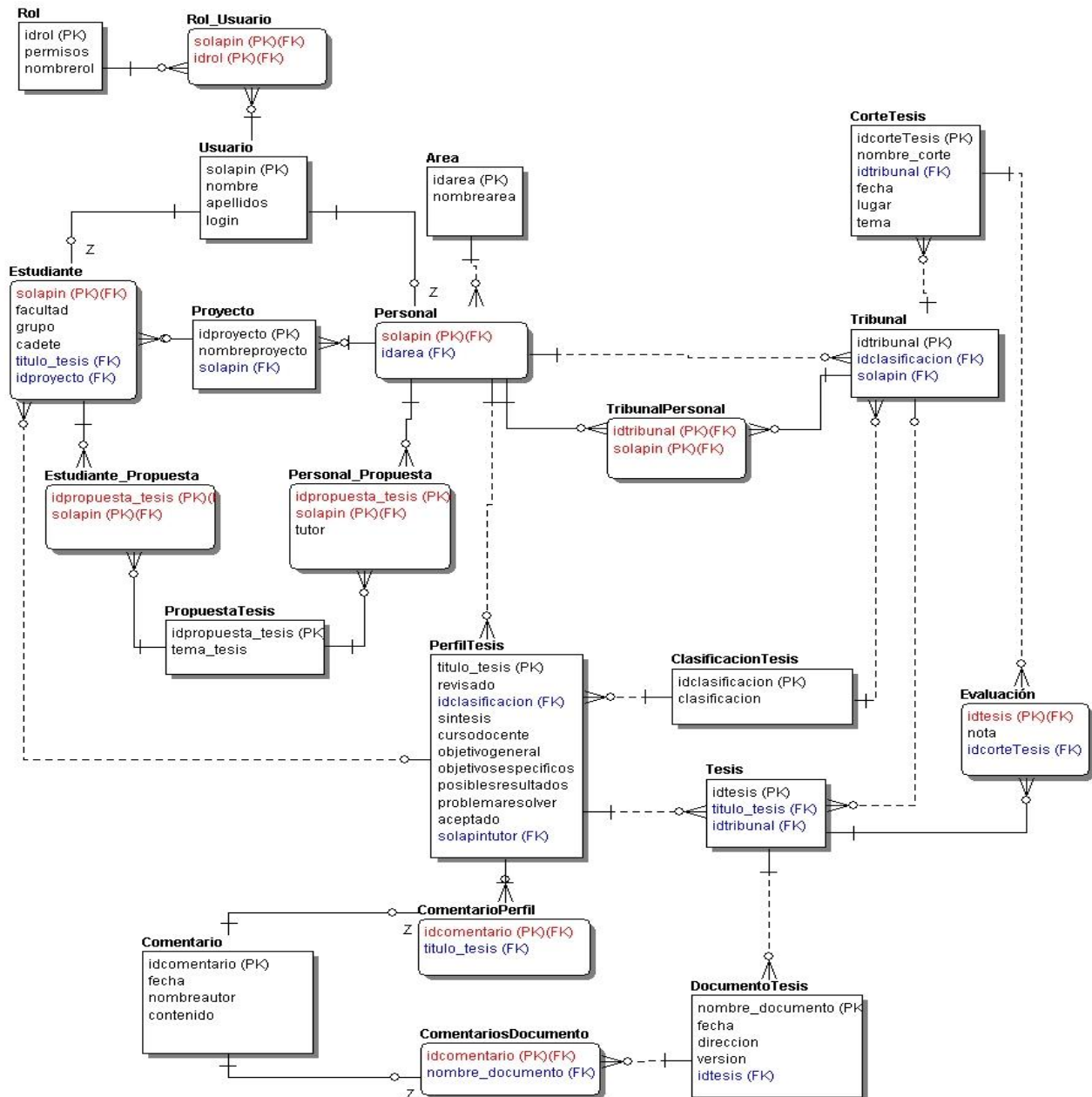


Figura 32: Modelo físico de datos

3.6 Patrón arquitectónico. Modelo-Vista-Controlador

Un patrón arquitectónico define a una familia de sistemas informáticos en términos de su organización estructural, describe componentes y las relaciones entre ellos con las restricciones de la aplicación, la composición asociada y el diseño para su construcción. Uno de los patrones más usados para desarrollar aplicaciones web es el Modelo-Vista-Controlador (MVC) (**Ver figura 3.30**), el cual es una aproximación al software que separa la lógica de la aplicación, de la presentación y, constituye además el patrón sobre el que se basa CodeIgniter, que fue el framework seleccionado para desarrollar esta aplicación. En la práctica, el MVC permite que sus páginas web contengan mínima codificación ya que la presentación es separada del código PHP. [27]

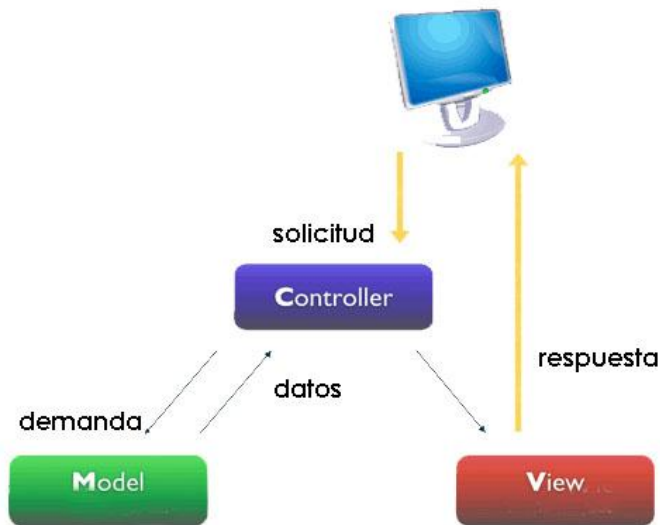


Figura 33: Interacción entre los componentes del patrón MVC

El Modelo representa la estructura de datos. Típicamente las clases del modelo contienen funciones que ayudan a recuperar, insertar y actualizar información en la base de datos.

La Vista es la información que es presentada al usuario. La Vista normalmente es una página web, pero en CodeIgniter específicamente, una vista también puede ser un fragmento de una página como un encabezado o un pie de página.

El Controlador sirve como un intermediario entre el Modelo, la Vista y cualquier otro recurso necesario para procesar la petición HTTP y generar una página web. [27]

Tanto la vista como el controlador dependen del modelo, el cual no depende de las otras clases. Esta separación permite construir y probar el modelo, independientemente de la representación visual.

Entre las ventajas del estilo Modelo-Vista-Controlador están las siguientes:

Soporte de múltiples vistas: Dado que la vista se halla separada del modelo y no hay dependencia directa del modelo con respecto a la vista, la interfaz de usuario puede mostrar múltiples vistas de los mismos datos simultáneamente. Por ejemplo, múltiples páginas de una aplicación Web pueden utilizar el mismo modelo de objetos mostrado de maneras diferentes.

Adaptación al cambio: Los requerimientos de interfaz de usuario tienden a cambiar con mayor rapidez que las reglas de negocios. Dado que el modelo no depende de las vistas, agregar nuevas opciones de presentación generalmente no afecta al modelo.

3.7 Tratamiento de errores

El tratamiento de los errores es un paso indispensable para el buen funcionamiento del sistema. Los errores más comunes que se pueden presentar están relacionados con el trabajo con la base de datos, fundamentalmente en los procesos de inserción, eliminación y/o modificación de datos. Es por esto que se han establecido mecanismos que visualicen la información para evitar en la medida de lo posible la introducción manual de los datos y así minimizar los errores que pueda tener la aplicación. Además se muestran mensajes indicando al usuario los errores cometidos, permitiéndole a la vez corregir su entrada con facilidad y continuar.

3.8 Seguridad

En el mundo de hoy donde la información constituye un elemento de vital importancia, es necesario que existan mecanismos de seguridad que garanticen la protección de los datos que se manipulan en la aplicación y que los usuarios según su rol solo puedan realizar las acciones que le atañen y por tanto obtener información únicamente de las tareas para las que tienen autorización.

A partir de esta idea, para mantener la seguridad de la información que se maneja, se tuvo en cuenta para el acceso al sitio, una autenticación previa, donde el sistema comprueba que tanto el usuario como la contraseña coincidan con las del dominio UCI y se le da acceso de acuerdo a los permisos que tiene asignado el rol correspondiente al usuario que accede. En caso contrario se re-direcciona a la página principal y se brinda la posibilidad de autenticarse nuevamente.

Además, el sistema está diseñado para garantizar la confiabilidad de la información, ya que se realizan diariamente copias de seguridad de tipo normal, tanto a las carpetas que contienen los documentos de tesis como a la base de datos.

3.9 Conclusiones

En este capítulo se definieron y describieron las clases y las relaciones entre ellas, a través de los diagramas de clases, tanto para el proceso de análisis como del diseño de la aplicación, lo cual permitió obtener una visión avanzada del sistema basada en la comprensión de los requisitos funcionales y no funcionales. Se modelaron además los diagramas de interacción correspondientes a los escenarios más significativos, de modo que quedó descrito más claramente el flujo de los mensajes entre los objetos del sistema y la dinámica de la aplicación. Finalmente se realizó el diseño de la base de datos, basando su desarrollo en la construcción de un modelo de datos, que tuvo su origen en el diagrama de clases persistentes.

Capítulo 4. Implementación

4.1 Introducción

En este capítulo se describen los elementos necesarios para la implementación, partiendo del resultado obtenido del diseño. Se muestra la distribución del sistema en nodos mediante el diagrama de despliegue y la organización de los componentes y las relaciones lógicas entre ellos a través del diagrama de componentes, quedando así conformado el modelo de implementación.

4.2 Diagrama de despliegue

El diagrama de despliegue que aparece a continuación muestra la disposición física de los distintos nodos que componen el sistema y el reparto de los componentes sobre dichos nodos, los cuales representan los objetos físicos con recursos computacionales. Las conexiones establecidas son asociaciones de comunicación entre los nodos, y se etiquetan con un estereotipo que identifica el protocolo de comunicación o la red utilizada.



Figura 34: Diagrama de despliegue del sistema

4.3 Diagramas de componentes

Los diagramas de componentes siguientes muestran un conjunto de elementos del modelo de implementación tales como componentes y sus relaciones.

Se utilizan para modelar la vista estática del sistema y muestra la organización y las dependencias lógicas entre los componentes software, sean éstos componentes de código fuente, librerías, binarios o ejecutables.

Caso de Uso: “Autenticar usuario”

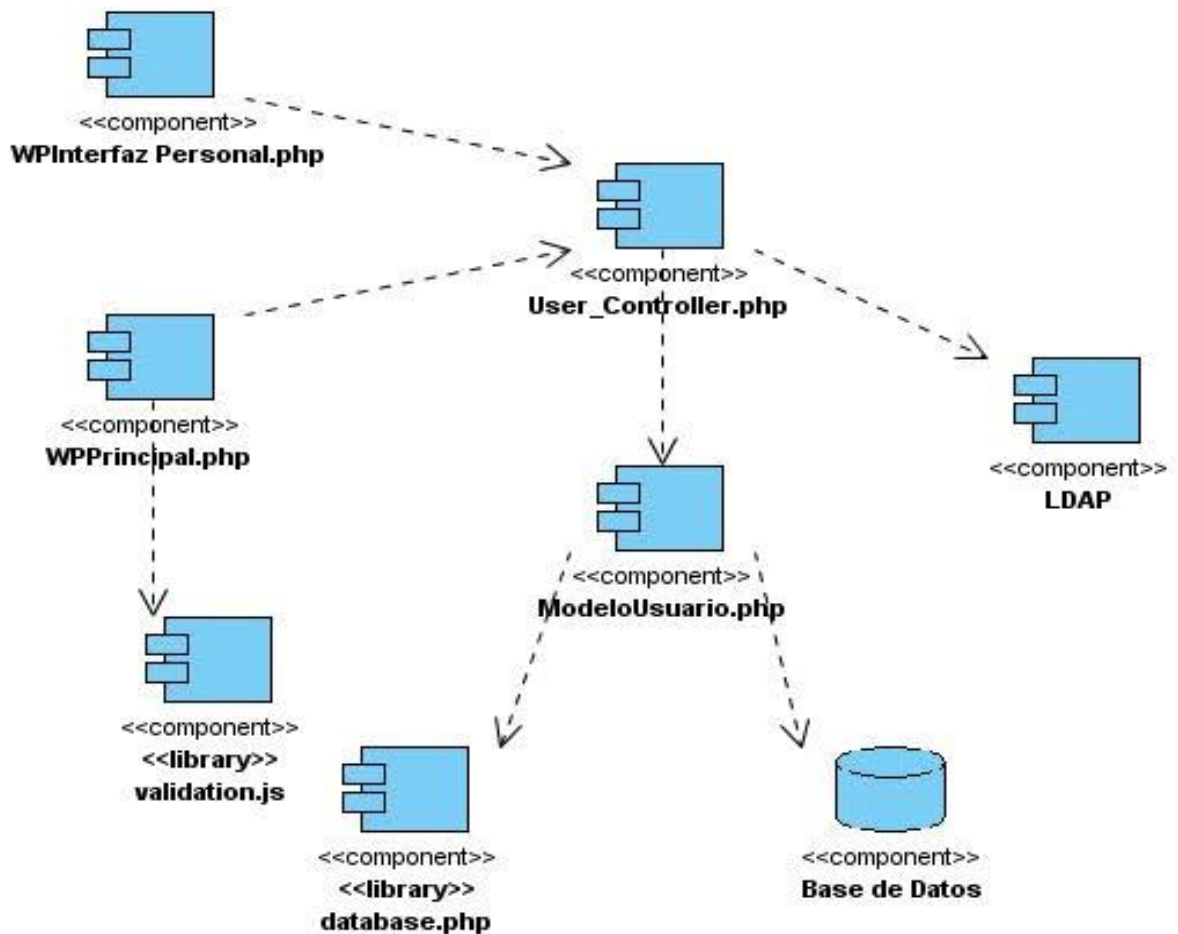


Figura 35: Diagrama de componentes del CU “Autenticar usuario”

Caso de Uso: "Evaluar perfil"

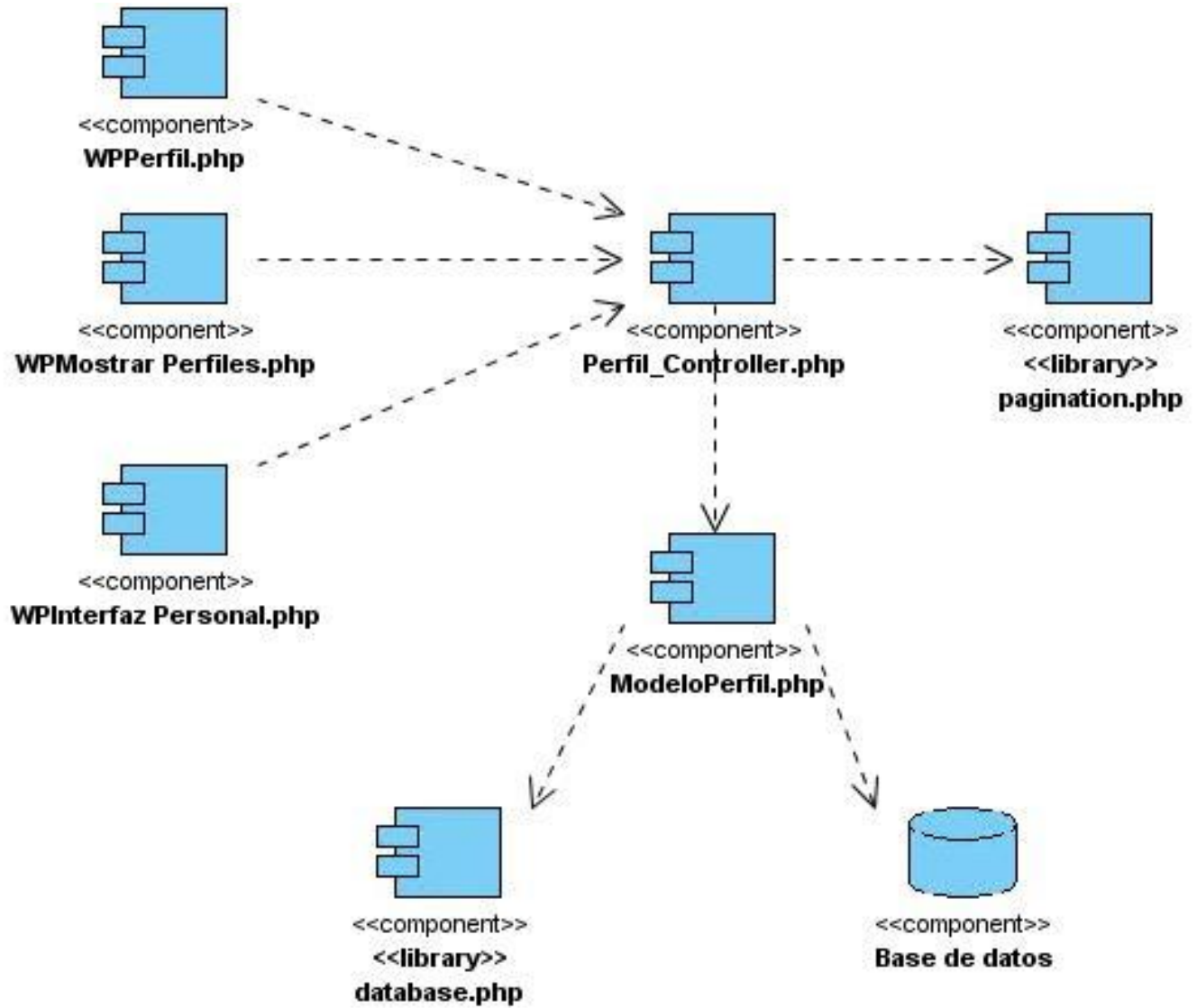


Figura 36: Diagrama de componentes del CU "Evaluar perfil"

Caso de Uso: “Gestionar propuestas de tesis”

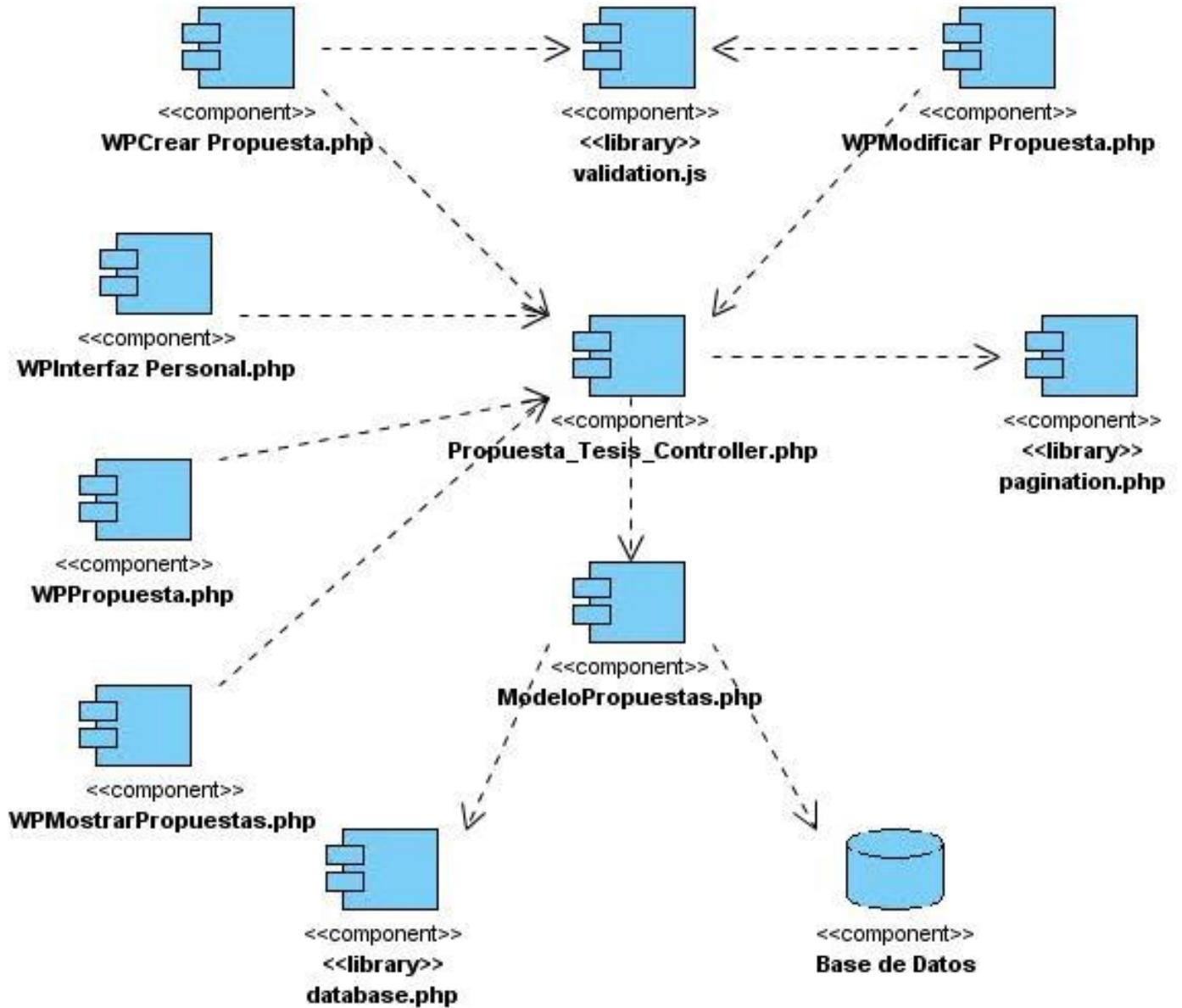


Figura 39: Diagrama de componentes del CU “Gestionar propuestas de tesis”

Caso de Uso: “Gestionar tribunal de tesis”

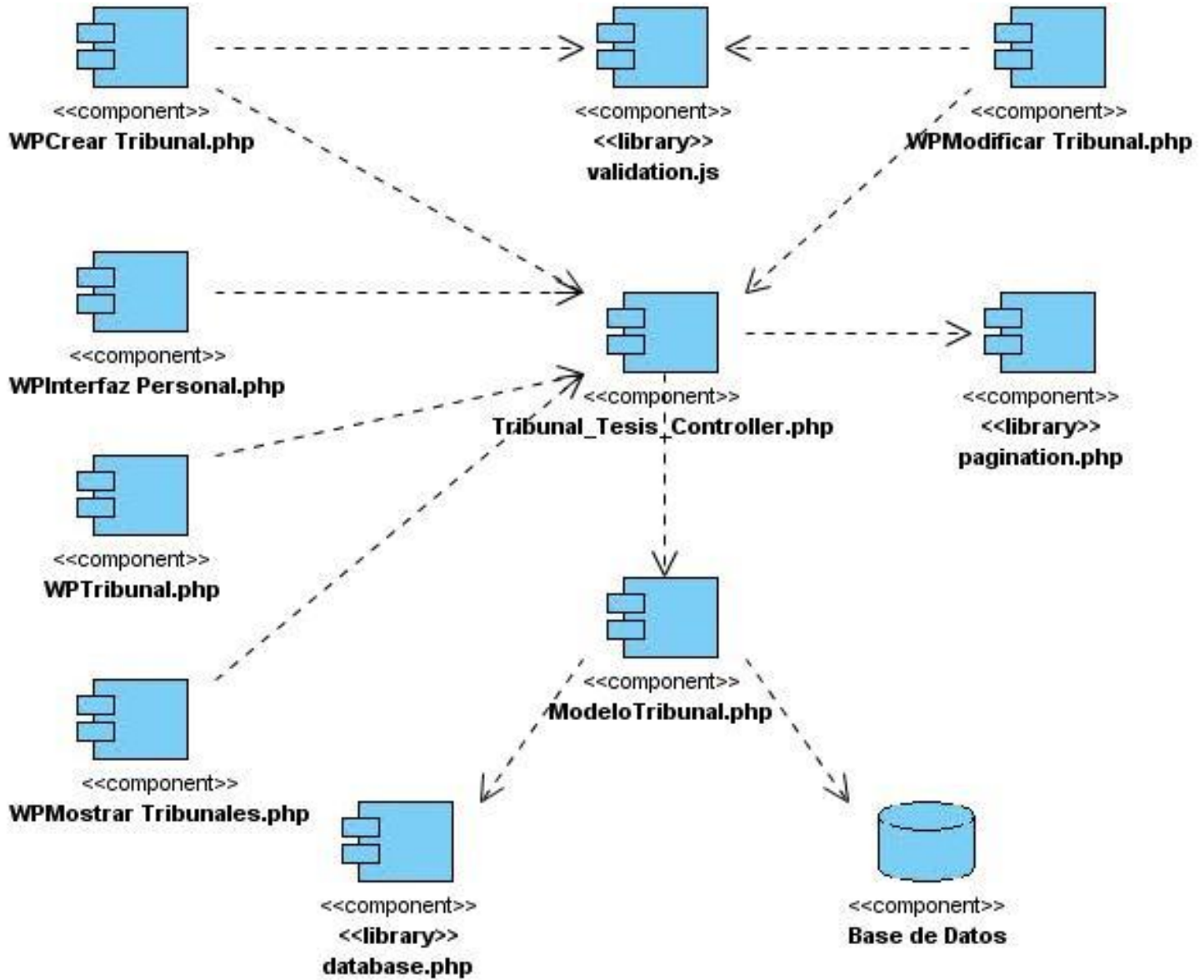


Figura 41: Diagrama de componentes del CU “Gestionar tribunal de tesis”

Caso de Uso: “Gestionar usuarios”

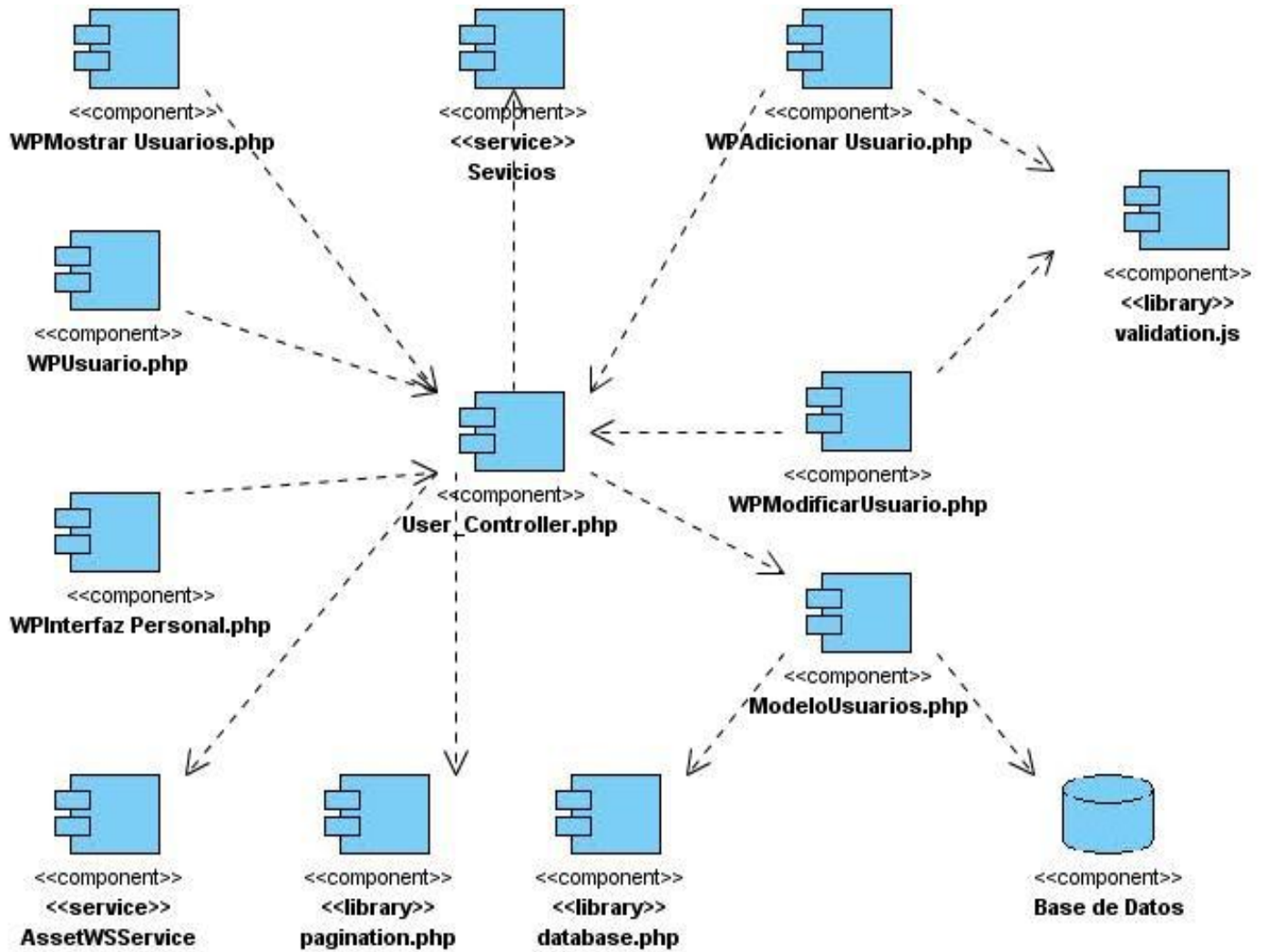


Figura 42: Diagrama de componentes del CU “Gestionar usuarios”

Caso de Uso: “Gestionar perfil de tesis”

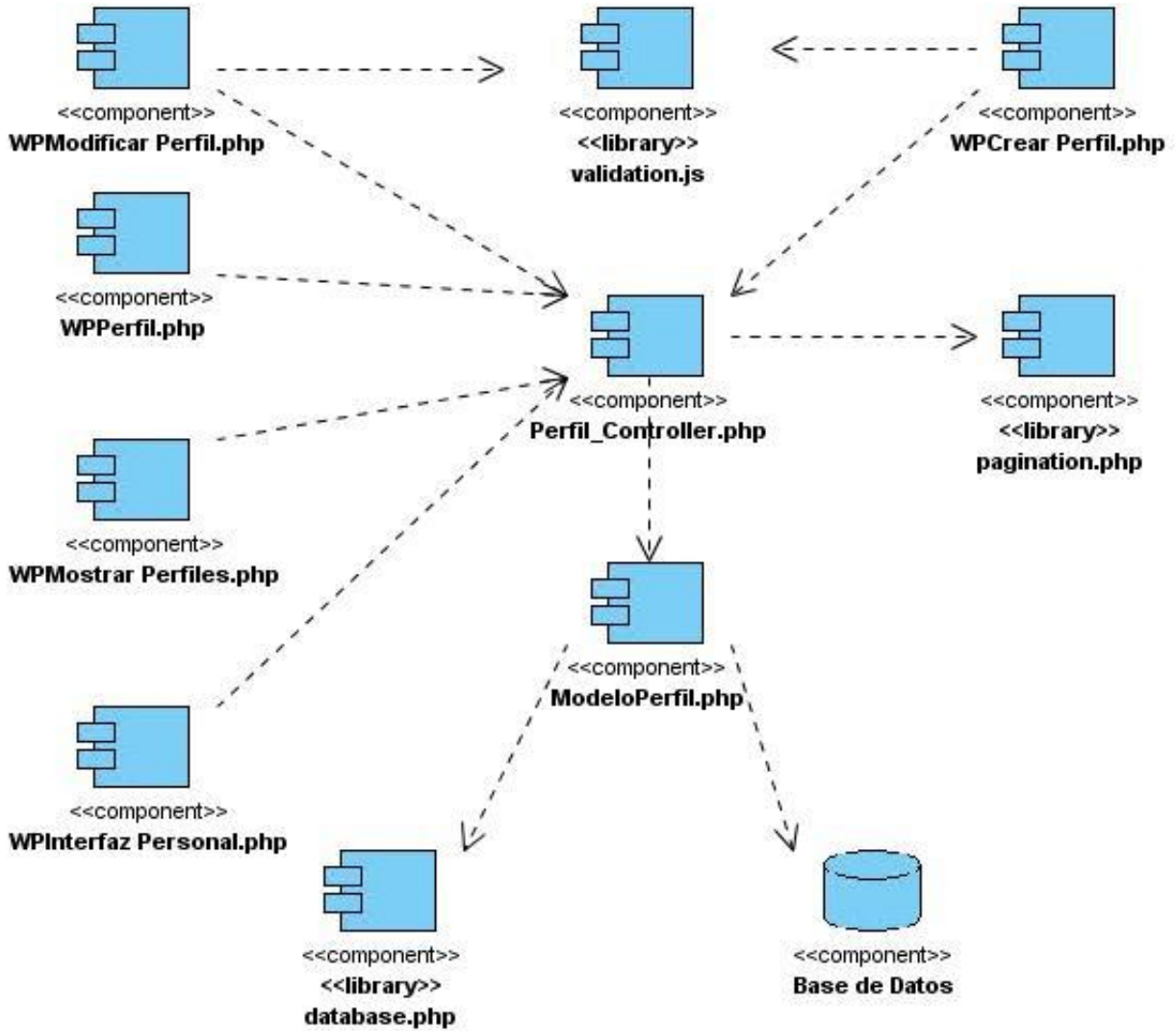


Figura 43: Diagrama de componentes del CU “Gestionar perfil de tesis”

4.4 Conclusiones

En este capítulo quedó conformado el modelo de implementación, ya que se realizó la distribución del sistema en nodos mediante el diagrama de despliegue y se mostró las organizaciones y las dependencias lógicas entre componentes a través de los diagrama de componentes.

Capítulo 5. Estudio de Factibilidad

5.1 Introducción

Para desarrollar un proyecto, es muy importante realizar un análisis profundo que permita determinar si es conveniente o no continuar con la elaboración del mismo. Para ello se realizó un estudio de la factibilidad, tras estimar los esfuerzos requeridos.

En el presente capítulo se detalla todo lo referente a este estudio, basado en la aplicación del método Análisis de Puntos de Casos de Uso. Como resultado del mismo se obtuvo el tiempo de desarrollo en meses y el costo aproximado, partiendo del hecho de que el proyecto sería desarrollado por dos personas.

5.2 Estimación por puntos de casos de uso

A pesar de que existen algunas alternativas posibles para la estimación del esfuerzo en proyectos basados en Casos de Uso, como es el caso del Análisis de Puntos de Función y COCOMO II, se decidió usar una variante más reciente denominada Análisis de Puntos de Casos de Uso, la cual es en cierta medida similar al Análisis de Puntos de Función. En el caso de la primera, la estimación es difícil de realizar si no se cuenta con una base histórica de proyectos que provea los coeficientes de conversión, además de que los valores estadísticos son difíciles de encontrar. Por su parte, COCOMO II, resulta muy útil para estimar un proyecto en forma global, cuando se tiene un conjunto de casos de uso bastante amplio y con escaso nivel de detalle, lo cual no se ajusta al caso en cuestión. Sin embargo, la estimación por Puntos de Caso de Uso resulta muy efectiva para estimar el esfuerzo requerido en el desarrollo de los primeros Casos de Uso de un sistema, si se sigue una aproximación iterativa como el Proceso Unificado de Rational, el cual coincide con la metodología de desarrollo aplicada al presente proyecto.

Este es 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 aplicación del mismo.

1- Cálculo de Puntos de Casos de Uso sin ajustar.

$$UUCP = UAW + UUCW$$

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.

1.1 Factor de Peso de los Actores sin ajustar (UAW)

Tipo de Actor	Descripción	Factor de peso	Actores	Total = (cant actores * peso)
Simple	Otro sistema que interactúa con el sistema a desarrollar mediante una interfaz de programación (API, Application Programming Interface)	1	0	1*0
Medio	Otro sistema que interactúa con el sistema a desarrollar mediante un protocolo o una interfaz basada en texto	2	1	1*2
Complejo	Una persona que interactúa con el sistema mediante una interfaz gráfica	3	5	5*3
TOTAL Σ				17

Tabla 4: Factor de Peso de los Actores sin ajustar (UAW)

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

Tipo de CU	Descripción	Peso	Cantidad de CU	Total = (cant CU * peso)
Simple	El caso de uso tiene de 1 a 3 transacciones.	5	1	5

Medio	El caso de uso tiene de 4 a 7 transacciones.	10	1	10
Complejo	El caso de uso tiene más de 8 transacciones.	15	7	105
TOTAL Σ				120

Tabla 5: Factor de Peso de los Casos de Uso sin ajustar (UUCW)

Por tanto:

$$UUCP = UAW + UUCW$$

$$UUCP = 17 + 120$$

$$UUCP = 137$$

2. Cálculo de Puntos de Casos de Uso ajustados.

$$UCP = UUCP \times TCF \times EF$$

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.

2.1 Factor de complejidad técnica (TCF).

Este coeficiente se calcula mediante la cuantificación de un conjunto de factores que determinan la complejidad técnica del sistema. Cada uno de los factores se cuantifica con un valor de 0 a 5, donde 0 significa un aporte irrelevante y 5 un aporte muy importante, esto está reflejado en la siguiente tabla:

Factor	Descripción	Peso	Valor asignado	Total = (peso* valor asignado)
T1	Sistema distribuido	2	0	0
T2	Tiempo de respuesta	1	2	2
T3	Eficiencia del usuario final	1	1	1
T4	Procesamiento interno complejo	1	0	0
T5	El código debe ser reutilizable	1	0	0
T6	Facilidad de instalación	0.5	3	1.5
T7	Facilidad de uso	0.5	4	2
T8	Portabilidad	2	4	8
T9	Facilidad de cambio	1	2	2
T10	Concurrencia	1	3	3
T11	Incluye objetivos especiales de seguridad	1	2	2
T12	Provee acceso directo a terceras partes	1	2	2
T13	Se requieren facilidades especiales de entrenamiento de usuarios	1	0	0
TOTAL = 0.6 + 0.01 * Σ				0.755

Tabla 6: Factor de complejidad técnica (TCF)

2.2 Factor de ambiente (EF).

Las habilidades y el entrenamiento del grupo involucrado en el desarrollo tienen un gran impacto en las estimaciones de tiempo. Estos factores son los que se contemplan en el cálculo del Factor de ambiente.

Factor	Descripción	Peso	Valor asignado	Total = (peso* valor asignado)
E1	Familiaridad con el modelo de proyecto utilizado	1.5	4	6
E2	Experiencia en la aplicación	0.5	4	2
E3	Experiencia en la orientación a objetivos	1	4	4
E4	Capacidad del analista líder	0.5	4	2
E5	Motivación	1	5	5
E6	Estabilidad de requerimientos	2	5	10
E7	Personal Part-Time.	-1	2	-2
E8	Dificultad del lenguaje de programación	-1	0	0
TOTAL = $1.4 - 0.03 * \sum$				0.59

Tabla 7: Factor de ambiente (EF)

- Para los factores del E1 al E4, un valor asignado de 0 significa sin experiencia, 3 experiencia media y 5 amplia experiencia (experto).
- Para el factor E5, 0 significa sin motivación para el proyecto, 3 motivación media y 5 alta motivación.
- Para el factor E6, 0 significa requerimientos extremadamente inestables, 3 estabilidad media y 5 requerimientos estables sin posibilidad de cambios.
- Para el factor E7, 0 significa que no hay personal part-time (es decir todos son full-time), 3 significa mitad y mitad, y 5 significa que todo el personal es part-time (nadie es full-time).

- Para el factor E8, 0 significa que el lenguaje de programación es fácil de usar, 3 medio y 5 que el lenguaje es extremadamente difícil.

Por tanto:

$$UCP = UUCP \times TCF \times EF$$

$$UCP = 137 \times 0.755 \times 0.59$$

$$UCP = 61.02665$$

3. De los puntos de casos de uso a la estimación.

- Se contabilizan cuántos factores de los que afectan al Factor de ambiente están por debajo del valor medio (3), para los factores del 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.
- Si el total es 2 o menos, se utiliza el factor de conversión 20 horas-hombre/Punto de Casos de Uso, es decir, un Punto de Caso de Uso toma 20 horas-hombre.
- Si el total es 3 o 4, se utiliza el factor de conversión 28 horas-hombre/Punto de Casos de Uso, es decir, un Punto de Caso de Uso toma 28 horas-hombre.
- Si el total es mayor o igual que 5, se recomienda efectuar cambios en el proyecto, ya que se considera que el riesgo de fracaso del mismo es demasiado alto.

Luego de haber realizado la comparación correspondiente en la tabla anterior se determinó que el factor de Conversión (CF) es de 20 horas/hombres.

El esfuerzo en horas-hombre está dado por:

$$E = UCP \times CF$$

Donde:

E: esfuerzo estimado en horas-hombre

UCP: Puntos de Casos de Uso ajustados

CF: factor de conversión

Luego:

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

$$E = 61.02665 \times 20$$

$$E = 1220.533 \sim 1221 \text{ horas/hombre}$$

Para una estimación más completa de la duración total del proyecto, se agrega a la obtenida por los Puntos de Casos de Uso, las estimaciones de esfuerzo de las demás actividades relacionadas con el desarrollo de software.

Actividad	% esfuerzo	Valor esfuerzo
Análisis	10	305.13325
Diseño	20	610.2665
Implementación	40	1220.533
Pruebas	15	457.699875
Sobrecarga (otras actividades)	15	457.699875
Total	100	3051.3325

Tabla 8: Estimaciones de esfuerzo por actividades

Suponiendo que una persona trabaje 8 horas por día, y tomando como promedio 30 días por mes; la cantidad de horas que puede trabajar una persona en 1 mes es 240 horas, de lo que se deduce la siguiente tabla.

Esfuerzo Total (Horas-- Hombre)	ET1	3051.3325
Esfuerzo Total (Mes-- Hombre)	ET2	12.71388541666666666666666666666667
Salario Promedio	SM	100
Cantidad de Hombres	CH	2
Costo (Hombre-Mes)	CHM	200
Costo Total	Costo	2542.7770833333333333333333333332

Tabla 9: Resumen de los resultados de la estimación

Finalmente:

$$\text{Tiempo} = \text{ET} / \text{CH}$$

$$\text{Tiempo} = 12.71388541666666666666666666666667 / 2$$

$$\text{Tiempo} = 6.35694270833333333333333333333333 \sim 6 \text{ meses}$$

De los resultados obtenidos se interpreta que con 2 hombres trabajando en el proyecto, el mismo se desarrolla en 6 meses y su costo total se estima sea de \$2542.77.

5.3 Beneficios tangibles e intangibles

La aplicación Tesis.Fac2 no es un producto con fines comerciales, ha sido creado con el objetivo de unificar en un espacio dedicado especialmente a las tesis, todo el contenido referente a las mismas.

El principal beneficio tangible que aporta el Sistema de Gestión de Tesis es el hecho de ser una aplicación Web dinámica, flexible y sencilla para actualizar y gestionar los datos necesarios para los servicios que brinda.

Referente a los beneficios intangibles, se pueden citar:

- ✓ Realiza la gestión de los tribunales de tesis, de modo que siempre se cuenta con la información real de los mismos, ya que las modificaciones se hacen sobre la aplicación.
- ✓ Brinda un punto de comunicación hacia los sistemas primarios de información (Intranet.uci.cu), los referentes al tema de las tesis (Tesis.uci.cu) y al correo.
- ✓ Aporta facilidad y rapidez en la revisión de los trabajos de tesis.
- ✓ Cuenta con un proceso de actualización de la información obtenida de los web services que se encuentran disponibles en la UCI.
- ✓ Optimiza el proceso de organización y desarrollo de los cortes evaluativos.

5.4 Análisis de costos

Para llevar a cabo el desarrollo de un sistema informático es necesario analizar los costos que incurrirán en su proceso.

Teniendo en cuenta las herramientas utilizadas para la realización del sistema, se puede decir que todas se pueden usar libremente, es decir, que no se producirán consumos en pagos de licencia.

Por tanto, analizando los costos y beneficios se puede llegar a la conclusión de que el desarrollo de este producto brindará más beneficios que costos, por lo que es factible su desarrollo y que su uso contribuirá a mejorar la gestión de las tesis en la Facultad 2 de la Universidad de la Ciencias Informáticas.

5.5 Conclusiones

En el presente capítulo se desarrolló la estimación por Puntos de Caso de Uso para estimar el esfuerzo del proyecto teniendo en cuenta los agentes que intervienen en el desarrollo del software. El tiempo y costo estimados para su realización, así como los beneficios que reportará la terminación del producto en cuestión, indican que es factible la implementación y utilización del mismo.

Conclusiones Generales

Luego de realizar un análisis profundo sobre la problemática existente en la Facultad 2 de la Universidad de las Ciencias Informáticas en cuanto al proceso de gestión de las tesis, se determinó llevar a cabo la automatización de los eventos relacionados con el tema. En el presente informe se detalla de forma clara cómo darle solución a la misma, describiendo las herramientas necesarias para implementar la aplicación, además de presentar un estudio de las tendencias actuales para desarrollar tareas similares. De forma general se cumplió satisfactoriamente el objetivo propuesto, ya que se creó el Sistema de Gestión de Tesis denominado "Tesis.Fac2".

Se obtuvieron además los siguientes resultados:

- 1- Quedó definida la metodología para la construcción del software a partir del estudio realizado.
- 2- Se generó toda la documentación detallada del análisis, diseño e implementación de la aplicación.
- 3- Quedó demostrada la validez y eficacia de las herramientas y tecnologías seleccionadas.
- 4- Se creó una base de datos con la información suficiente respecto a cada estudiante, tesis y personal involucrado en el asesoramiento de las mismas.

Por lo anteriormente descrito queda reflejado el cumplimiento de los objetivos específicos.

Recomendaciones

Con el objetivo de mejorar el sistema de gestión Tesis.Fac2, se recomienda:

- ✓ Extender las funcionalidades del sistema de forma tal que también se pueda gestionar toda la información referente a las evaluaciones de los cursos de posgrado.
- ✓ Agregar funcionalidades que permitan realizar reclamaciones, por parte de los estudiantes, respecto a las evaluaciones recibidas en los cortes de tesis y, de igual modo, que el presidente del tribunal al que pertenezca, pueda responder dichas reclamaciones.
- ✓ Incluir entre el grupo de reportes que se brindan, un informe general en el que se muestre un resumen de los datos más significativos por cada corte realizado, agrupado por tribunales.
- ✓ Permitir que se generen boletines informativos con noticias de interés, relacionadas con el proceso de tesis.
- ✓ Generalizar la propuesta en aras de que sea utilizada por otras facultades que no cuenten con un sistema que gestione la información que aquí se trata.
- ✓ Gestionar los permisos de los roles de usuarios de modo que estos se almacenen en la base de datos, para incrementar el dinamismo de la aplicación en este sentido.
- ✓ De forma general, dar continuidad y profundizar en los procesos de gestión de tesis.

Referencias bibliográficas

1. ¿Qué son los sistemas de gestión? [En línea] 2009. <http://www.bsigroup.com.mx/es-mx/Auditoria-y-Certificacion/Sistemas-de-Gestion/De-un-vistazo/Que-son-los-sistemas-de-gestion/>
2. Tesis. La etimología de la palabra tesis. [En línea] 03 09, 2009. <http://dieumsnh.qfb.umich.mx/gesinfo/tesis.htm#etimologiatesis1>.
3. **Hernández Meléndrez, Edelsys.** Metodología de la investigación. Como escribir una tesis. [En línea] 2006. <http://www.sld.cu/galerias/pdf/sitios/cirugiamaxilo/guiatesis.pdf.pdf>.
4. ¿Qué es un ERP? [En línea] 24 04 ,2006 <http://www.mundosap.com/foro/showthread.php?t=424>
5. La gestión documental de las tesis doctorales en la universidad de Castilla-la Mancha (2003). [En línea] 01 22, 2008. <http://www.uclm.es/archivo/paginas/difusion5.asp>
6. **Montero, M., García Peñalvo, A., García Carrasco, F., & Alonso Romero, J. (s.f.).** Dpto. Informática y Automática. Facultad de Ciencias. Universidad de Salamanca. Dpto. Teoría e Historia de la Educación. Facultad de Ciencias. Universidad de Salamanca. [En línea] Componentes software para entornos virtuales de educación.
7. **Mesa Martinez, Marcel. 2008.** *Tesis: Arquitectura de la capa de presentación para el proyecto CICPC.* 2008. Pág 8.
8. **Mesa Martinez, Marcel. 2008.** *Tesis: Arquitectura de la capa de presentación para el proyecto CICPC.* 2008. Pág 7
9. El lenguaje HTML. [En línea] 2008-2009. <http://www.juntadeandalucia.es/averroes/iesgaviota/informatica/html.html>.

10. Guía breve de CSS. [En línea] 01 09, 2008. <http://www.w3c.es/divulgacion/guiasbreves/HojasEstilo>
11. ¿Qué es javascript? http://74.125.47.132/search?q=cache:-HlgGFkkwXAJ:geneura.ugr.es/~victor/cursillos/javascript/js_intro.html+javascript+es+orientado+a+eventos&cd=3&hl=es&ct=clnk&gl=cu
12. Introducción a AJAX. [En línea] 01 15, 2009. <http://www.librosweb.es/ajax/>.
13. Introducción a AJAX. Capítulo 4. DOM (Document Object Model)) [En línea] 01 20, 2009 <http://www.librosweb.es/ajax/>
14. ¿Qué es un Servidor Web? [En línea] 2007-2008. <http://casidiablo.net/%C2%BFque-es-un-servidor-web/>.
15. **Pereira Rosa, Manuel.** Instituto Superior Pedagógico “Rafael María de Mendive” de Pinar del Río. Caracterización de una tipología de ejercicios para el proceso de enseñanza del Modelo Entidad/relación. [En línea]. <http://www.pr.rimed.cu/sitios/Revista%20Mendive/Num18/pdf/8%20Manuel%20Pereira.pdf>.
16. **J Kabir, Mohammed.** *Servidor Apache 2. Pág 42*
17. Características de PHP. [En línea] 04 06, 2009. <http://www.linuxcentro.net/linux/staticpages/index.php?page=CaracteristicasPHP>
18. ¿Qué es la Web 2.0? [En línea] 10 27, 2005. <http://www.maestrosdelweb.com/editorial/web2/>.
19. Búsqueda de Términos [En línea] 2008 <http://www.tecnolinkinformatica.com/definicion/Framework>
20. Web oficial de CodeIgniter. [En línea] 04 30, 2008. <http://codeigniter.com>.

21. **Solis Álvarez, Camilo Javier and Figueroa Díaz, Roberth Gustavo.** Universidad Técnica Particular de Loj. Metodologías Tradicionales vs. Metodologías Ágiles. [En línea]
22. Metodología. Metodologías rígidas y metodologías ágiles. [En línea] 2005.
<http://www.encamina.com/boletines/ENCAMINA%20y%20las%20metodolog%C3%ADas%20software.htm>.
23. **Jacobson, I., Booch, G., & Rumbaugh, J.** El proceso unificado de desarrollo del software. Capítulo 1. 2000. Pág 5.
24. **Hernández González, Anaisa.** Identificación de procesos de negocio [En línea] 12, 2004.
<http://www.cujae.edu.cu/ediciones/Revistas/Industrial/Vol-XXV/3-2004/83-88%20Identificaci%C3%B3n%20de%20procesos.pdf>
25. **Larman Craig.** UML y Patrones. Introducción al análisis y diseño orientado a objetos. Primera edición 1999. Pág. 187.
26. Unidad IV: Introducción a la programación. [En línea] 04 06, 2007 <http://ingunderworld.bloginom.com/>
27. CodeIgniter User Guide Version 1.7.1. 2006-2008.

Bibliografía

- *¿Qué es la Web 2.0?* [En línea] 27 de octubre de 2005. Obtenido de:
<http://www.maestrosdelweb.com/editorial/web2/>
- *¿Qué son las bases de datos?* [En línea] 26 de octubre de 2007. Obtenido de
<http://www.maestrosdelweb.com/principiantes/%c2%bfque-son-las-bases-de-datos/>
- *¿Qué son los sistemas de gestión?* [En línea] 2009. Obtenido de <http://www.bsigroup.com.mx/es-mx/Auditoria-y-Certificacion/Sistemas-de-Gestion/De-un-vistazo/Que-son-los-sistemas-de-gestion/>
- Alvarez, S. [En línea] 30 de agosto de 2007. *Arquitectura cliente-servidor*. Obtenido de
<http://www.desarrolloweb.com/articulos/arquitectura-cliente-servidor.html>
- *Capítulo 1: Introducción a JavaScript*. [En línea] 2008. Obtenido de
<http://www.elcodigo.net/tutoriales/javascript/javascript1.html>
- *CodeIgniter User Guide Version 1.7.1*. [En línea] 2006-2008.
- *CSS desde cero*. [En línea] 20 de enero de 2009. Obtenido de <http://css.infames.org/ventajas.html>
- de la torre, Anibal [En línea] 2006. *Lenguajes del lado servidor o cliente*. Obtenido de
http://www.adelat.org/media/docum/nuke_publico/lenguajes_del_lado_servidor_o_cliente.html
- *El lenguaje HTML*. [En línea] 2008-2009. Obtenido de
<http://www.juntadeandalucia.es/averroes/iesgaviota/informatica/html.html>
- *Guía breve de CSS*. [En línea] 09 de enero de 2008. Obtenido de
<http://www.w3c.es/divulgacion/guiasbreves/HojasEstilo>

- Gutiérrez, J. J. (s.f.). *¿Qué es un framework web?* Obtenido de http://www.lsi.us.es/~javierj/investigacion_ficheros/Framework.pdf
- Hernández Meléndrez, E. [En línea] 2006. *Metodología de la investigación. Cómo escribir una tesis*. Obtenido de <http://www.sld.cu/galerias/pdf/sitios/cirugiamaxilo/guiatesis.pdf.pdf>)
- *Introducción a AJAX*. [En línea] 15 de enero de 2009. Obtenido de <http://www.librosweb.es/ajax/>
- J Kabir, M. *Servidor Apache 2*.
- Jacobson, I., Booch, G., & Rumbaugh, J. [En línea] 1998. *El lenguaje unificado de modelado. Manual de referencia*.
- Jacobson, I., Booch, G., & Rumbaugh, J. [En línea] 2000. *El proceso unificado de desarrollo del software*.
- JMPereda. [En línea] 24 de agosto de 2007. *¿Qué es una Aplicacion Web?* Obtenido de <http://jimpereda.wordpress.com/2007/08/24/definiendo-la-plantilla/>
- Juansa. [En línea] 2009. *Introducción a Redes. Arquitectura cliente/servidor*.
- *La gestión documental de las tesis doctorales en la universidad de Castilla-la Mancha (2003)*. [En línea] 22 de enero de 2008. Obtenido de <http://www.uclm.es/archivo/paginas/difusion5.asp>
- Larman, C. [En línea] 1999. *UML y Patrones. Introducción al análisis y diseño orientado a objetos*.
- *Los Frameworks de PHP agilizan tu trabajo*. [En línea] 31 de julio de 2007. Obtenido de <http://www.maestrosdelweb.com/editorial/los-frameworks-de-php-agilizan-tu-trabajo/>

- Mesa Martínez, M. [En línea] 2008. *Tesis: Arquitectura de la capa de presentación para el proyecto CICPC.*
- *Metodología. Metodologías rígidas y metodologías ágiles.* [En línea] 2005. Obtenido de <http://www.encamina.com/boletines/ENCAMINA%20y%20las%20metodolog%C3%ADas%20software.htm>
- *Modelo Cliente Servidor.* (s.f.). Obtenido de <http://sipan.inictel.gob.pe/users/hherrera/mcs.htm>
- Montero, M., García Peñalvo, A., García Carrasco, F., & Alonso Romero, J. (s.f.). *Componentes software para entornos virtuales de educación.*
- Pereira Rosa, M. (s.f.). *Caracterización de una tipología de ejercicios para el proceso de enseñanza del Modelo Entidad/relación.* Obtenido de <http://www.pr.rimed.cu/sitios/Revista%20Mendive/Num18/pdf/8%20Manuel%20Pereira.pdf>
- *SCS - Sistemas Cliente-Servidor y Procesos Cooperativos.* [En línea] 03 de octubre de 2008. Obtenido de <http://ccia.ei.uvigo.es/docencia/SCS>
- *Sistemas gestores de bases de datos.* [En línea] 31 de julio de 2007. Obtenido de <http://www.desarrolloweb.com/articulos/sistemas-gestores-bases-datos.html>
- Solís Álvarez, C. J., & Figueroa Díaz, R. G. (s.f.). *Metodologías Tradicionales vs. Metodologías Ágiles.*
- *Tecnología cliente servidor.* (s.f.). Obtenido de <http://www.inei.gob.pe/biblioineipub/bancopub/inf/lib5038/indice.HTM>

- *Tesis. La etimología de la palabra tesis.* [En línea] 09 de marzo de 2009. Obtenido de Tesis. La etimología de la palabra tesis
- *Web oficial de CodeIgniter.* [En línea] 30 de abril de 2008. Obtenido de <http://codeigniter.com>
- *¿Qué es un servidor Web?* [En línea] 2007-2008. Obtenido de <http://casidiablo.net/%C2%BFque-es-un-servidor-web/>
- *¿Qué es un ERP?* [En línea] 24 de abril de 2006. Obtenido de <http://www.mundosap.com/foro/showthread.php?t=424>
- *A partir de hoy MySQL es propiedad de Sun Microsystems.* [En línea] 16 de enero de 2008. Obtenido de <http://www.maestrosdelweb.com/actualidad/a-partir-de-hoy-mysql-es-propiedad-de-sun-mycrosystems/>
- *¿Qué es javascript?* Obtenido de http://74.125.47.132/search?q=cache:-HlgGFkkwXAJ:geneura.ugr.es/~victor/cursillos/javascript/js_intro.html+javascript+es+orientado+a+eventos&cd=3&hl=es&ct=clnk&gl=cu
- *Búsqueda de Términos.* [En línea] 2008. Obtenido de <http://www.tecnolinkinformatica.com/definicion/Framework>
- **Hernández González, Anaisa.** [En línea] diciembre de 2004. *Identificación de procesos de negocio.* Obtenido de <http://www.cujae.edu.cu/ediciones/Revistas/Industrial/Vol-XXV/3-2004/83-88%20Identificaci%C3%B3n%20de%20procesos.pdf>
- *Unidad IV: Introducción a la programación.* [En línea] 4 de junio de 2007. Obtenido de <http://ingunderworld.bloginom.com/>

- *Desarrollo ERP/CRM.* [En línea] 18 de septiembre de 2005. Obtenido de <http://desarrolloerp.blogspot.com/2005/09/erp-cubano-opensource.html>
- *Características de PHP.* [En línea] 4 de junio de 2009. Obtenido de <http://www.linuxcentro.net/linux/staticpages/index.php?page=CaracteristicasPHP>

Anexos

Anexo 1. Descripción textual de los Casos de Uso del Negocio en formato expandido.

Caso de Uso 1: “Realizar Tesis”.

Nombre del caso de uso:	Realizar Tesis
Actores del negocio:	Estudiante
Propósito:	Los estudiantes deben desarrollar su tesis bajo la supervisión de los tutores y deberán ser sometidos a cortes evaluativos para garantizar la calidad del trabajo.
Resumen:	El caso de uso se inicia cuando el estudiante comienza a desarrollar su tesis, la cual es sometida por parte de los tutores a revisiones periódicas previas a los cortes de tesis hasta arribar a la defensa final, de modo que el trabajo tenga un mínimo de errores.
Casos de Uso asociados	Asignar Perfil de Tesis.
Precondiciones	El estudiante debe tener un perfil de tesis aprobado.
Curso normal de los eventos	
Acción del actor	Respuesta del negocio
1. Solicita un tema de tesis	2. El comité de tesis selecciona un tema de tesis. 3. El comité de tesis busca un profesor disponible para tuturar. 4. Se efectúa el caso de uso asociado “Asignar Perfil de Tesis”.

5. El estudiante comienza a desarrollar su tesis.	7. El tutor revisa el trabajo y confirma al estudiante que no hay errores. Finaliza el caso de uso.
6. El estudiante envía su trabajo para ser revisado.	
Curso alterno de los eventos	
Acción del actor	Respuesta del negocio
7.3 El estudiante corrige su trabajo y retorna al paso 6.	7.1 En caso de haber correcciones el tutor realiza los señalamientos pertinentes. 7.2 El tutor envía las correcciones realizadas.
Pos condiciones	

Tabla 10: Descripción del CUN “Realizar Tesis”

Caso de Uso 2: “Asignar Perfil de Tesis”.

Nombre del caso de uso:	Asignar Perfil de Tesis.
Actores del negocio:	Estudiante
Propósito:	Asignar un perfil de tesis a los estudiantes.
Resumen:	El caso de uso se inicia cuando el comité de tesis o el líder de un determinado proyecto asignan un tema de tesis con su respectivo tutor a un estudiante dado.
Casos de Uso asociado	
Precondiciones	El estudiante debe haber aprobado la prueba de nivel de Programación.

Curso normal de los eventos	
Acción del actor	Respuesta del negocio
	<ol style="list-style-type: none"> 1. El comité de tesis de la facultad o el líder de proyecto informa tanto al estudiante como al tutor de la propuesta de tesis asignada. 2. El tutor elabora el perfil de tesis usando o no la propuesta recibida y lo envía al comité de tesis y a los estudiantes involucrados.
3. El estudiante revisa el perfil de tesis.	
	4. El comité de tesis aprueba el perfil de tesis, envía la confirmación y finaliza el caso de uso.
Curso alterno de los eventos	
Acción del actor	Respuesta del negocio
	<ol style="list-style-type: none"> 4.1 En caso de no ser aprobado el perfil de tesis, el comité de tesis se lo devuelve al tutor. 4.2 El tutor corrige el perfil de tesis. 4.3 Se retorna al paso 2. 4.4 Finaliza el caso de uso.
Poscondiciones	El estudiante debe tener un perfil de tesis aprobado con su respectivo tutor, y cotutor en caso de ser necesario.

Tabla 11: Descripción del CUN “Asignar Perfil de Tesis”

Caso de Uso 3: “Realizar Corte de Tesis”.

Nombre del caso de uso:	Realizar Corte de Tesis	
Actores del negocio:	Estudiante	
Propósito:	Que los estudiantes desarrollen su tesis bajo un seguimiento y control de todas las mismas, desde su inicio hasta la discusión final.	
Resumen:	El caso de uso se inicia cuando el Comité de Tesis planifica las comisiones y lo notifica a los estudiantes, los cuales acuden en la fecha indicada para realizar su exposición y ser evaluados de acuerdo al trabajo realizado.	
Casos de Uso asociados		
Precondiciones	Que el estudiante haya empezado a desarrollar su tesis.	
Curso normal de los eventos		
Acción del actor	Respuesta del negocio	
	<ol style="list-style-type: none"> 1. El Comité de Tesis conforma las comisiones. 2. El Comité de Tesis planifica las fechas de exposiciones. 3. El Comité de Tesis notifica la planificación al estudiante. 	
4. El estudiante acude en la fecha indicada al corte de tesis.	<ol style="list-style-type: none"> 5. El tribunal encargado evalúa el trabajo presentado por el estudiante. 6. El tribunal emite una nota final a la exposición y la comisión emite un informe del corte de 	

	tesis y finaliza el Caso de Uso.
Pos condiciones:	El estudiante debe tener una evaluación en el corte y debe quedar conformado un informe con los datos del mismo.

Tabla 12: Descripción del CUN “Realizar Corte de Tesis”

Anexo 2. Diagramas de Actividades

Caso de Uso “Realizar Tesis”

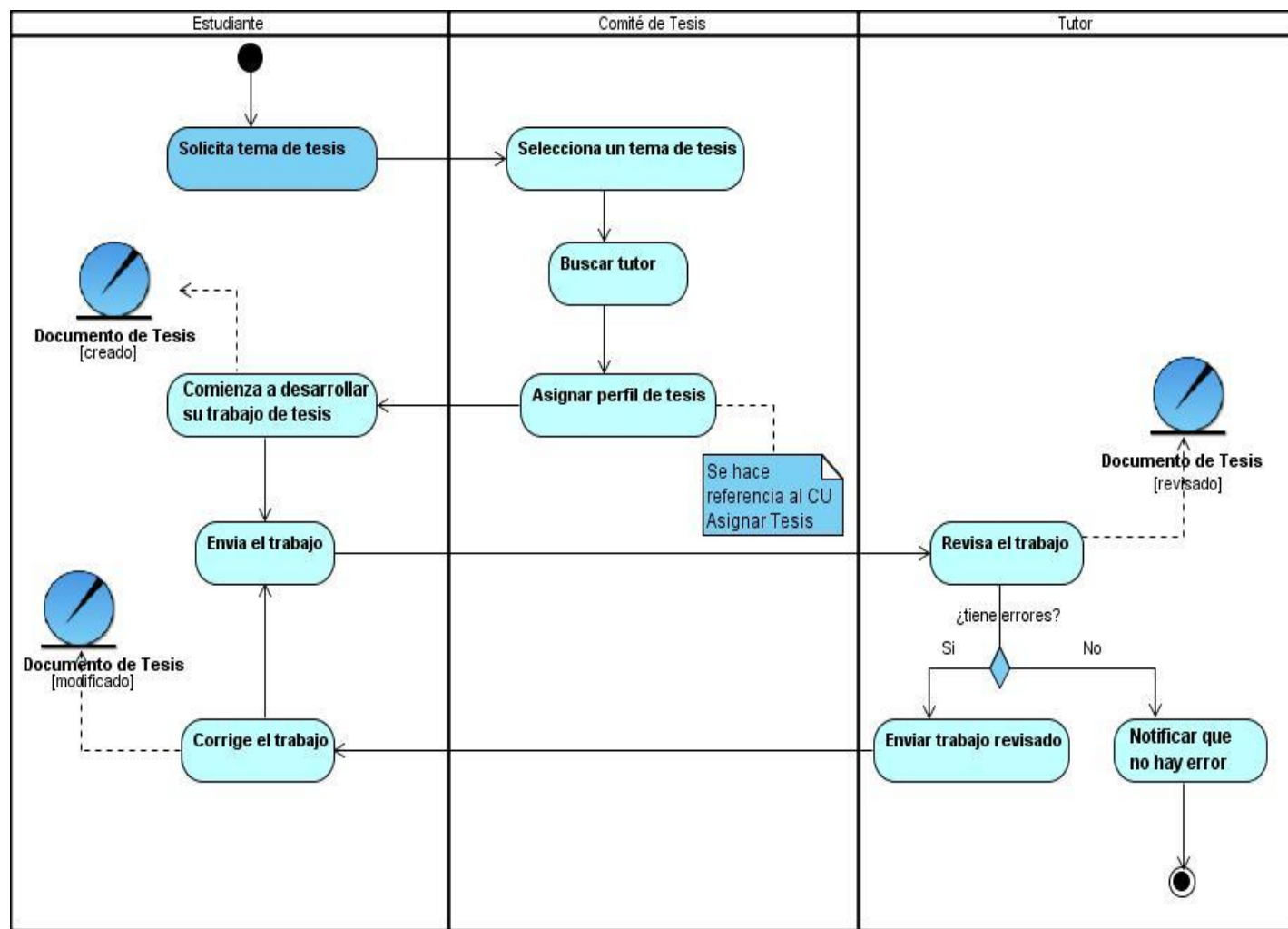


Figura 44: Diagrama de actividades del CUN “Realizar tesis”

Caso de Uso "Asignar Perfil de Tesis"

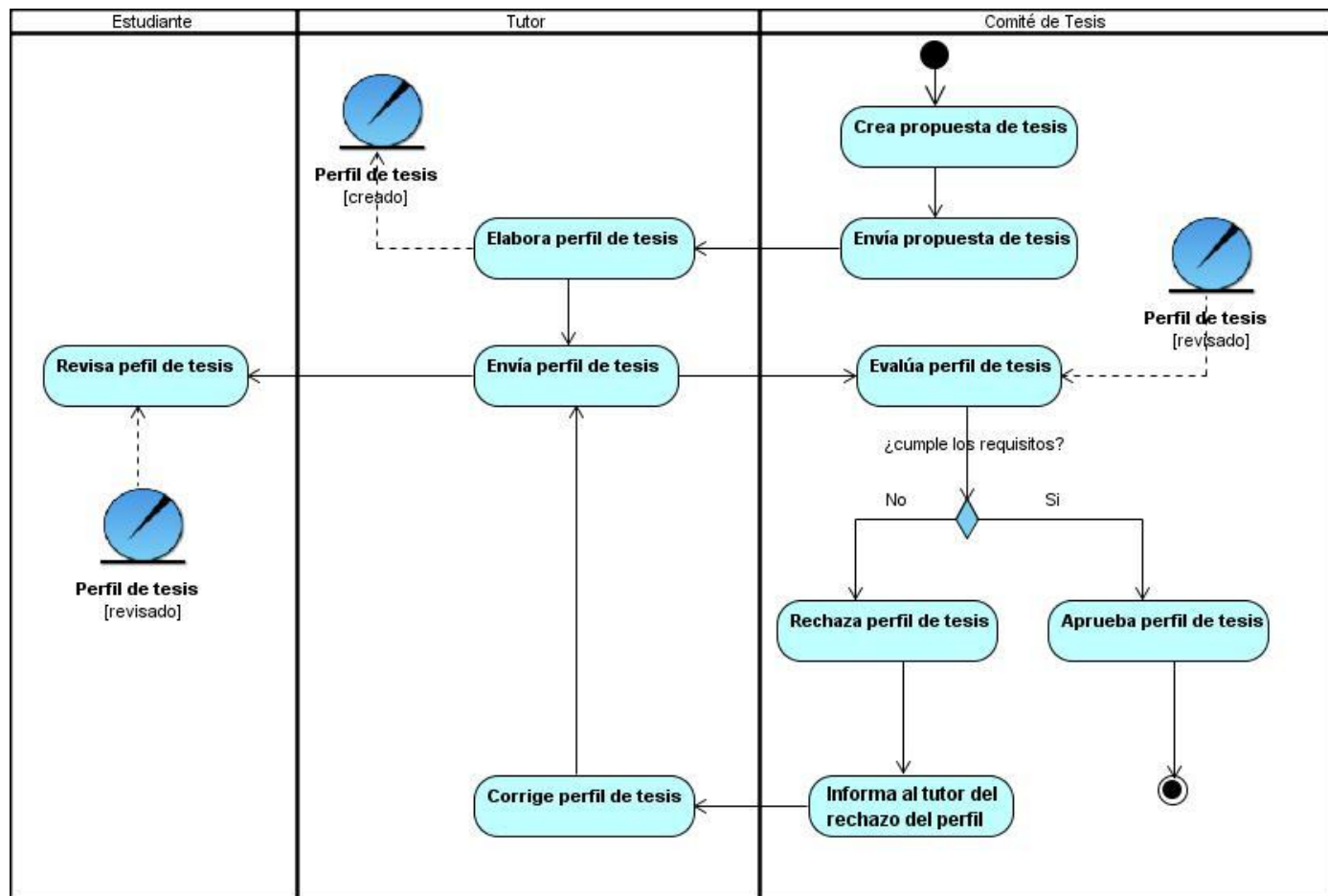


Figura 45: Diagrama de actividades del CUN "Asignar perfil de tesis"

Caso de Uso "Realizar Corte de Tesis"

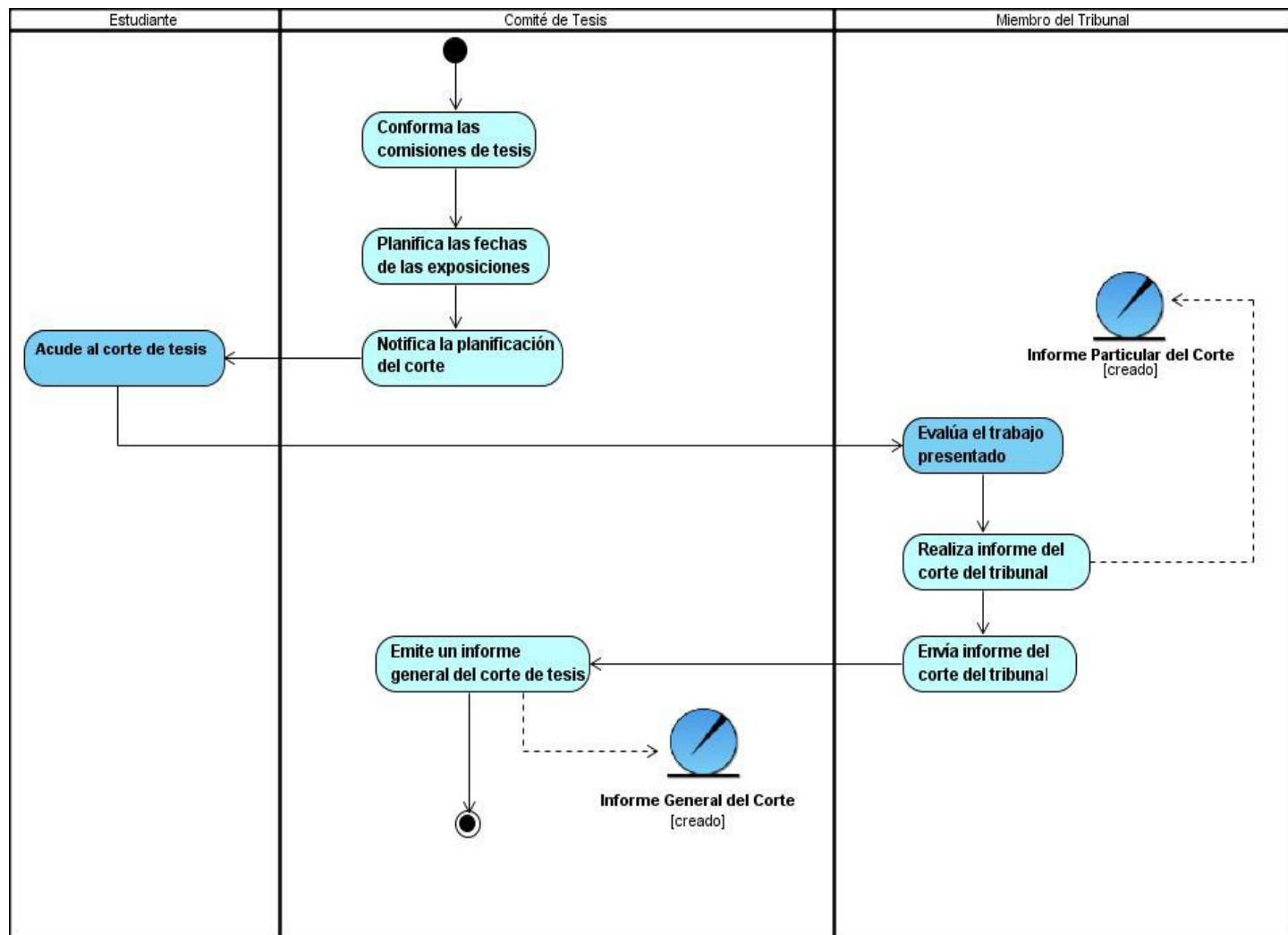


Figura 46: Diagrama de actividades del CUN "Realizar corte de tesis"

Anexo 3. Modelo de objetos

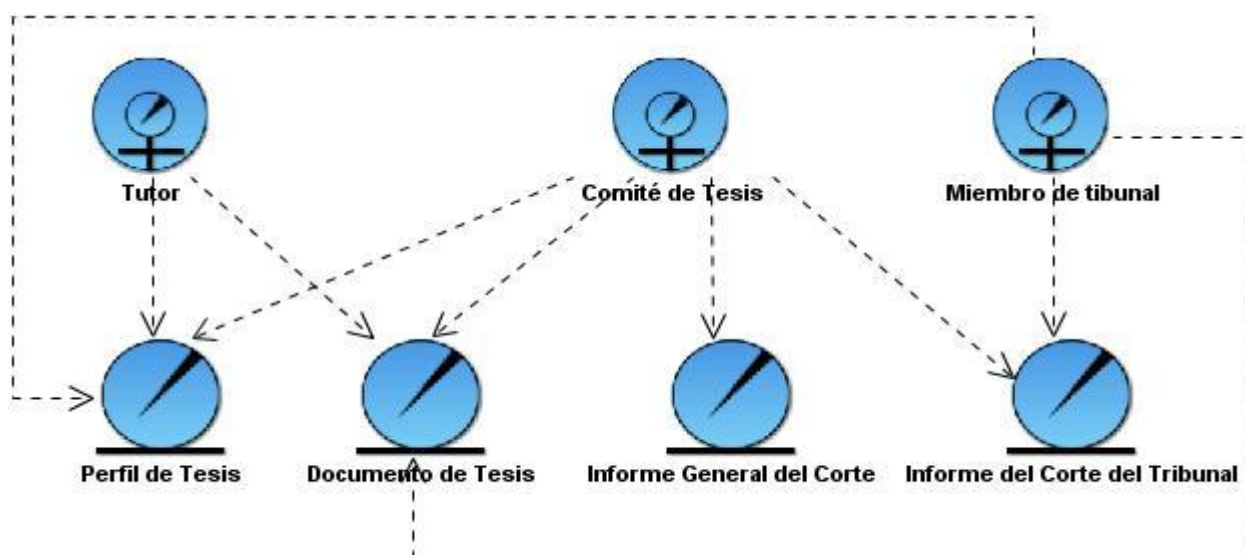


Figura 47: Modelo de objetos

Anexo 4. Descripción textual de los Casos de Uso del Sistema

Caso de Uso 1: “Autenticar Usuario”

Caso de Uso:	Autenticar Usuario	
Actores:	Usuario	
Resumen:	El caso de uso se inicia cuando el Usuario introduce en el sistema su usuario y contraseña, y se le da acceso en dependencia del rol que desempeñe.	
Precondiciones:	Debe tener una cuenta en el dominio UCI.	
Referencias:	RF1	
Prioridad:	Crítico	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Sistema	

<p>1. El Usuario introduce su usuario y su contraseña del dominio UCI y presiona el botón “Entrar” en la opción “Iniciar sesión”</p>	<p>1.1 El sistema verifica que los datos sean correctos y muestra sobre la interfaz el listado de roles que tiene asignados.</p>
<p>2. El Usuario selecciona el rol por el que desea acceder al sistema.</p>	<p>1.2 El sistema muestra una interfaz con el menú correspondiente al rol seleccionado y finaliza el caso de uso.</p>

Flujos Alternativos de Eventos

Acción del Actor	Respuesta del Sistema
	<p>1.1 Si los datos introducidos por el usuario son incorrectos se un mensaje de error.</p> <p>1.2 Se retorna al paso 1.</p>

	
<p>Intranet Inter-nos Tesis-UCI Correo FEU Home</p>	
<p>Inicio de Sesión</p> <p>Usuario (UCI) <input type="text"/></p> <p>Password <input type="password"/></p> <p><input type="button" value="Entrar"/></p> <p>Usuario o Contraseña incorrecta, verifiquelo.</p>	<p>Un sistema completamente automatizado para la gestión de las tesis en la facultad #2</p> <p>Personalizado a la medida de sus necesidades. Todas sus responsabilidades están bajo su control.</p> 
<p>Poscondiciones</p>	<p>Debe quedar definido el nivel de acceso del Usuario.</p>

Tabla 13: Descripción del CUS “Autenticar Usuario”

Caso de Uso 2: “Gestionar Documento de Tesis”

Caso de Uso:	Gestionar Documento de Tesis
Actores:	Estudiante, Tutor
Resumen:	El caso de uso se inicia cuando el estudiante accede al sistema para cargar un documento de tesis y así esté disponible para que las personas autorizadas lo revisen y emitan un criterio de él. El tutor acceder para

	cargar los documentos de las tesis que supervisa, con las correcciones realizadas.
Precondiciones:	El actor debe estar autenticado en el sistema y el documento debe estar en formato Word y con un límite de 3MG.
Referencias:	RF8
Prioridad:	Crítico
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El actor accede a la opción del menú “Mis documentos”.	1.1 El sistema muestra la interfaz correspondiente para el control de los documentos de tesis, enviando a dicha interfaz el listado de los documentos de tesis guardados hasta el momento.
2. El actor selecciona la acción a realizar. En caso de: <ol style="list-style-type: none"> 1. “Cargar nuevo documento de tesis” ir a sección 1. 2. “Descargar documento de tesis” ir a sección 2. 3. “Eliminar documento” ir a descripción del caso de uso 17 “Eliminar documento”. 	
Sección “Cargar nuevo documento de tesis”	
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El actor presiona el botón Nuevo documento.	1.1 Levanta una interfaz para indicar la dirección donde se encuentra el documento.
2. El actor presiona el botón “Browse” o “Cargar”.	2.1 El sistema muestra un “Open Dialog” para especificar la ruta completa donde se encuentra el documento brindando las opciones de “Abrir” o “Cancelar”.
3. El actor selecciona el documento a cargar y oprime “Abrir”.	3.1 El sistema verifica la extensión y tamaño del documento, lo almacena en un directorio y retorna un mensaje de confirmación finalizando el caso de uso.
Flujos Alternativos de eventos	
Acción del Actor	Respuesta del Sistema

3. El actor presiona el botón "Cancelar".	3.1 El sistema desactiva el "Open Dialog" y retorna a la interfaz finalizando el caso de uso.
	3.1 En caso del documento no tener una extensión válida o exceder el tamaño límite, se envía un mensaje de error, retorna al paso 2 y finaliza el caso de uso.
Sección "Descargar documento de tesis"	
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El actor selecciona el documento que desea descargar y presiona el botón Descargar.	1.1 Levanta una ventana para indicar la dirección donde desea guardar el documento.
2. El actor escoge la dirección donde desea guardar el documento y presiona el botón Guardar.	2.1 Descarga el documento en la dirección señalada. 2.2 Vuelve a la interfaz inicial y termina el caso de uso.
Sección "Eliminar documento de tesis"	
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. Selecciona el documento que desea eliminar y presiona el botón Eliminar.	1.1 Muestra un mensaje de confirmación para saber si el usuario esta seguro de querer eliminar el documento seleccionado.
2. Presiona el botón Aceptar.	2.1 Elimina el documento. 2.2 Retorna a la interfaz anterior con la actualización del documento eliminado y finaliza el caso de uso.
Flujos Alternativos de Eventos	
Acción del Actor	Respuesta del Sistema
2. Presiona el botón Cancelar.	2.1 Retorna a la interfaz anterior y finaliza el caso de uso.

Intranet Inter-nos Tesis-UCI Correo FEU Home (Salir)

Opciones del Usuario

Notificar corte

Notificar tribunales

Notificar perfil

Mis documentos

Listado de Documentos de Tesis guardados

Nombre Documento	Fecha de guardado	Acciones	Comentarios
Referencias.doc	2009-05-27	  	Revisar
Referencias1.doc	2009-05-27	  	Revisar

Subir

Usted está registrado como Maite Sosa Veranes. Desde el IP: 10.7.8.30 Page rendered in 0.0985 seconds

Tabla 14: Descripción del CUS “Gestionar Documento de Tesis”

Caso de Uso 3: “Gestionar Perfil de Tesis”

Caso de Uso:	Gestionar Perfil de Tesis
Actores:	Tutor.
Resumen:	El caso de uso se inicia cuando el tutor toma los datos de la propuesta recibida y elabora el perfil de tesis correspondiente al tema seleccionado. De esta forma el perfil una vez creado puede ser editado y eliminado.
Precondiciones:	
Referencias:	RF 7
Prioridad:	Crítico
Flujo Normal de Eventos	

Acción del Actor	Respuesta del Sistema
1. El tutor accede a la opción del sistema "Perfiles de Tesis".	1.1 El sistema muestra los perfiles creados por el tutor y la opción de "Crear un nuevo perfil". Así como "Ver detalles" de los perfiles creados. En caso de: <ol style="list-style-type: none"> 1. "Crear un nuevo perfil" ir a la sección 1. 2. "Ver detalles" ir a la sección 2. 3. "Eliminar perfil" ir a sección 3. 4. "Realizar cambios" ir a sección 4.



Intranet Inter-nos Tesis-UCI Correo FEU Home

(Salir)

Opciones del Usuario

Listado de perfiles de tesis

- Notificar corte
- Notificar tribunales
- Evaluar Tesis
- Mis Perfiles
- Documentos de Tesis

Titulo de Tesis	Aceptado	Acciones	Comentario
Esto es una prueba nueva con la BD nueva.	Si		Revisar
Gestion de riesgos de erp	Si		Revisar
Sistema de Gestion de Tesis para la Facultad 2	Si		Revisar

1 2 >

Nuevo perfil

Usted está registrado como Marcel Mesa Martinez. Desde el IP: 10.7.8.30

Page rendered in 0.7133 seconds

Sección 1 "Crear perfil de tesis"

Flujo Normal de Eventos

Acción del Actor	Respuesta del Sistema
1. El tutor presiona el botón "Crear un nuevo perfil".	1.1 El sistema muestra un formulario con un listado de todos los estudiantes que aún no tienen perfil de tesis asignado y los campos correspondientes para llenar los datos.

2. El tutor llena los campos y presiona el botón "Enviar".	2.1 El sistema valida los datos entrados y en caso de estar correctos y completos, almacena estos en la base de datos y retorna a la primera interfaz terminando el caso de uso.
Sección 2 "Ver detalles del perfil"	
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El tutor presiona el botón "Ver detalles".	1.1 El sistema muestra una interfaz con todos los datos del perfil de tesis seleccionado y finaliza el caso de uso.
Sección 3 "Eliminar perfil de tesis"	
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1 El tutor presiona el botón "Eliminar perfil".	1.2 El sistema muestra un mensaje de diálogo para "Confirmar" o "Cancelar" la acción.
2 El tutor oprime "Confirmar".	2.1 El sistema elimina toda la información referente al perfil de tesis, se retorna un mensaje informando que se ha eliminado y finaliza el caso de uso.
Flujos Alternativos de Eventos	
Acción del Actor	Respuesta del Sistema
2. El tutor oprime "Cancelar".	2.1 El sistema se mantiene en la misma interfaz y finaliza el caso de uso.
Sección 4 "Realizar cambios"	
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El tutor modifica los datos del perfil y presiona el botón "Enviar cambios".	1.1 El sistema verifica los datos y los guarda en la base de datos. 1.2 Muestra un mensaje de confirmación y finaliza en el caso de uso.

Flujos Alternativos de Eventos	
Acción del Actor	Respuesta del Sistema
1.2 Retorna al paso 1 y finaliza el caso de uso.	1.1 En caso de haber datos incorrectos o incompletos se envía un mensaje de error.

Tabla 15: Descripción del CUS “Gestionar Perfil de Tesis”

Caso de Uso 4: “Gestionar tribunal de tesis”

Caso de Uso:	Gestionar tribunal de tesis
---------------------	-----------------------------

Actores:	Comité de Tesis.	
Resumen:	El caso de uso se inicia cuando el Comité de Tesis accede al sistema para conformar los distintos tribunales de tesis para los cortes, pre defensas y defensas finales. De esta forma se crean los tribunales de acuerdo a las clasificaciones de las tesis, ya sean de desarrollo o investigativas.	
Precondiciones:		
Referencias:	RF 5	
Prioridad:	Crítico	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Sistema	
1. El comité de tesis accede al menú y selecciona la opción "Tribunales de tesis".	1.1 Muestra la interfaz correspondiente, enviando a la misma el listado de todos los tribunales creados hasta el momento y brinda todas las posibles acciones a ejecutar.	
2. El actor selecciona la acción a realizar. En caso de: <ul style="list-style-type: none"> 1. "Crear tribunal de tesis" ir a sección 1. 2. "Ver detalles" ir a sección 2. 3. "Eliminar tribunal" ir a sección 3. 4. "Modificar los datos del tribunal" ir a sección 4. 		



Tesis.Fac2
Sistema de Gestión de Tesis

Intranet Inter-nos Tesis-UCI Correo FEU Home (Salir)

Opciones del Usuario

- Cortes de Tesis
- Tribunales de Tesis
- Mis Perfiles
- Evaluar perfil
- Clasificaciones de Tesis

Listado de tribunales de tesis

Seleccione	Claficiacion	Jefe del Tribunal	Acciones
<input type="checkbox"/>	Antivirus	ARIAGNIS YERO GUEVARA	✖ 📄 🔍
<input type="checkbox"/>	ERP	Marcel Mesa Martánez	✖ 📄 🔍
<input type="checkbox"/>	ENUM	LEX KAREL ZAYAS HERNANDEZ	✖ 📄 🔍

Eliminar todos
Nuevo tribunal

Usted está registrado como Yadira Ruiz Constanten. Desde el IP: 10.7.8.30 Page rendered in 0.2139 seconds

Sección 1 “Crear tribunal de tesis”

Flujo Normal de Eventos

Acción del Actor	Respuesta del Sistema
1. El miembro de comité de tesis oprime el botón “Crear tribunal de tesis”.	1.1 El sistema muestra un formulario con los campos a llenar y el listado de los profesores disponibles para integrar el nuevo comité, dando la opción de seleccionar a un profesor como presidente, así como las tesis que se evaluarán en ese tribunal.
2. El comité de tesis completa los campos y selecciona los profesores integrantes, así como cuál de ellos será el presidente.	2.1 El sistema verifica los datos entrados y se crea un nuevo tribunal que es almacenado en la BD. 2.2 Envía un mensaje de confirmación y finaliza el caso de uso.

Flujos Alternativos de Eventos

Acción del Actor	Respuesta del Sistema

	2.1 En caso de encontrarse errores o incompletos los datos, se muestra un mensaje de error indicando los campos incorrectos y se retorna al paso 2 finalizando el caso de uso.
Sección 2 “Ver detalles del tribunal”	
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El comité de tesis selecciona un tribunal y presiona el botón “Ver detalles”.	1.1 El sistema muestra una interfaz con todos los datos del tribunal de tesis seleccionado.
Sección 3 “Eliminar tribunal”	
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1 El miembro del comité presiona el botón “Eliminar tribunal”.	1.1 El sistema muestra un mensaje de diálogo para “Confirmar” o “Cancelar” la acción.
2 El miembro del comité oprime “Confirmar”.	2.1 El sistema elimina toda la información referente al tribunal de tesis, se retorna un mensaje informando que se ha eliminado y finaliza el caso de uso.
Flujos Alternativos de Eventos	
Acción del Actor	Respuesta del Sistema
2. El miembro del comité oprime “Cancelar”.	2.1 El sistema se mantiene en la misma interfaz y finaliza el caso de uso.
Sección 4 “Modificar datos del tribunal de tesis”	
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
2. El miembro del comité modifica los datos del tribunal y presiona el botón “Enviar cambios”.	1.1 El sistema verifica los datos y los guarda en la base de datos. 1.2 Muestra un mensaje de confirmación y finaliza en el caso de uso.
Flujos Alternativos de Eventos	
Acción del Actor	Respuesta del Sistema


1.2 Retorna al paso 1 y finaliza el caso de uso.	1.1 En caso de haber datos incorrectos o incompletos se envía un mensaje de error.
--	--

Tabla 16: Descripción del CUS “Gestionar tribunal de tesis”

Caso de Uso 5: “Evaluar perfil de tesis”

Caso de Uso:	Evaluar Perfil de Tesis	
Actores:	Comité de Tesis.	
Resumen:	El caso de uso se inicia cuando el Comité de Tesis accede al sistema para revisar los perfiles de tesis que no han sido marcados como revisados ni están aprobados. De esta forma revisa cada perfil y lo marca como Aprobado o Rechazado, a la vez que automáticamente luego de ser enviados estos perfiles se marcan los mismos como revisados. De esta forma se establece una secuencia de envíos y revisiones necesarias hasta que se corrijan los defectos de cada perfil y para poder ser aprobados.	
Precondiciones:		
Referencias:	RF 4	
Prioridad:	Crítico	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Sistema	
1. El miembro del comité de tesis accede a la opción del “Perfiles de tesis”.	1.1 El sistema muestra un listado de los perfiles de tesis que han sido creados, señalando los que están aprobados y lo que no.	
2. El miembro del comité de tesis selecciona uno de los perfiles mostrados.	1.3 El sistema muestra una interfaz con los datos correspondientes al perfil seleccionado y brinda las opciones de “Aprobar”, “Rechazar” y “Agregar comentario”	
3. En caso de: 1. “Aprobar perfil de tesis” ir a la sección 1. 2. “Rechazar perfil de tesis” ir a la sección 2. 3. “Agregar comentario” ir a sección 3.		

Sección 1 “Aprobar perfil de tesis”	
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El miembro del comité de tesis oprime el botón “Aprobar”.	1.1 El sistema marca como Aprobado el perfil de tesis. 1.2 Se crea una nueva tesis con los datos necesarios del perfil que ha sido aprobado y se almacena en la base de datos. 1.3 Almacena los datos del perfil de tesis y muestra un mensaje de confirmación finalizando el caso de uso.
Sección 2 “Rechazar perfil de tesis”	
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El miembro del comité de tesis oprime el botón “Rechazar”.	1.1 El sistema marca el perfil de tesis como Rechazado y finaliza el caso de uso.
Sección 3 “Agregar comentario”	
Flujos Alternativos de Eventos	
Acción del Actor	Respuesta del Sistema
1. Ir a la descripción el caso de uso 12 “Realizar comentarios”.	







Intranet Inter-nos Tesis-UCI Correo FEU Home (Salir)

Opciones del Usuario

- Cortes de Tesis
- Tribunales de Tesis
- Mis Perfiles
- Evaluar perfil
- Clasificaciones de Tesis

Listado de perfiles de tesis

Titulo de Tesis	Aceptado	Acciones
Esto es una prueba nueva con la BD nueva.	Si	
Gestion de riesgos de erp	Si	
Sistema de Gestion de Tesis para la Facultad 2	Si	
Este es el ultimo que mando Mayte de Maylin Fundora.	Si	

Usted está registrado como Yadira Ruiz Constanten. Desde el IP: 10.7.8.30
Page rendered in 0.1538 seconds

Tabla 17: Descripción del CUS “Evaluar perfil de tesis”

Caso de Uso 6: “Gestionar proyectos”

Caso de Uso:	Gestionar proyectos
Actores:	Administrador
Resumen:	El caso de uso se inicia cuando el administrador accede al sistema para controlar la información referente a los proyectos y así lograr un control eficiente de la asignación de tesis a los estudiantes vinculados a proyectos.
Precondiciones:	
Referencias:	RF 2
Prioridad:	Secundario

Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El administrador accede a la opción del menú "Proyectos".	1.1 Muestra la interfaz correspondiente, enviando a la misma el listado de todos los proyectos creados hasta el momento y brinda todas las posibles acciones a ejecutar.
2. El administrador selecciona la acción a realizar. En caso de: <ol style="list-style-type: none"> 1. "Adicionar proyecto" ir a sección 1. 2. "Ver detalles del proyecto" ir a sección 2 3. "Eliminar proyecto" ir a sección 3 4. "Modificar los datos del proyecto" ir a sección 4. 	

Opciones del Usuario

- Proyectos
- Roles de usuarios
- Usuarios
- Areas

Listado de proyectos

Seleccione	Nombre Proyecto	Líder de proyecto	Acciones
<input type="checkbox"/>	ERP	BARBARA TRIANA MORALES	✖ 📄 🔍
<input type="checkbox"/>	ENUM	ARIAGNIS YERO GUEVARA	✖ 📄 🔍
<input type="checkbox"/>	Antivirus	NIOBIS VALIENTE LOPEZ	✖ 📄 🔍

Eliminar todos **Nuevo proyecto**

Usted está registrado como Administrador. Desde el IP: 10.7.8.30 Page rendered in 0.1876 seconds

Sección 1 "Adicionar proyecto"

Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema

	2.1 En caso de haber datos incorrectos o incompletos se muestra un mensaje de error indicando los campos incorrectos y se retorna al paso 2 finalizando el caso de uso.
Sección 2 “Ver detalles del proyecto”	
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El administrador selecciona un proyecto y presiona el botón “Ver detalles”.	1.1 El sistema muestra una interfaz con todos los datos del proyecto seleccionado.
Sección 3 “Eliminar proyecto”	
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El administrador presiona el botón “Eliminar proyecto”.	1.1 El sistema muestra un mensaje de diálogo para “Confirmar” o “Cancelar” la acción.
2. El administrador oprime “Confirmar”.	2.1 El sistema elimina toda la información referente al proyecto, se retorna un mensaje informando que se ha eliminado y finaliza el caso de uso.
Flujos Alternativos de Eventos	
Acción del Actor	Respuesta del Sistema
2. El administrador oprime “Cancelar”.	2.1 El sistema se mantiene en la misma interfaz y finaliza el caso de uso.
Sección 4 “Modificar datos del proyecto”	
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El administrador modifica los datos del proyecto y presiona el botón “Enviar cambios”.	1.1 El sistema verifica los datos y los guarda en la base de datos. 1.2 Muestra un mensaje de confirmación y finaliza en el caso de uso.
Flujos Alternativos de Eventos	
Acción del Actor	Respuesta del Sistema

1.2 Retorna al paso 1 y finaliza el caso de uso.	1.1 En caso de haber datos incorrectos o incompletos se envía un mensaje de error.
--	--

Tabla 18: Descripción del CUS “Gestionar proyectos”

Caso de Uso 7: “Gestionar propuestas de tesis”

Caso de Uso:	Gestionar propuestas de tesis
Actores:	Líder de proyecto y Comité de tesis
Resumen:	El caso de uso se inicia cuando los miembros del comité de tesis o el líder de un proyecto acceden al sistema para realizar las propuestas de tesis a los estudiantes y así lograr un control eficiente de la asignación de tesis a los mismos.
Precondiciones:	
Referencias:	RF 3
Prioridad:	Crítico
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El actor accede a la opción del menú “Propuestas de tesis”.	1.1 Muestra la interfaz correspondiente, enviando a la misma el listado de todas las propuestas creadas hasta el momento y brinda todas las posibles acciones a ejecutar.
2. El actor selecciona la acción a realizar. En caso de: 1. “Adicionar propuesta de tesis” ir a sección 1. 2. “Ver detalles de la propuestas de tesis” ir a sección 2. 3. “Eliminar propuesta de tesis” ir a sección 3. 4. “Modificar los datos de la propuesta de tesis” ir a sección 4.	
Sección 1 “Adicionar propuesta de tesis”	

Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El actor oprime el botón “Adicionar propuesta de tesis”.	1.1 El sistema muestra una interfaz con un formulario que contiene los campos a llenar, dados por el listado de los estudiantes a los que aun no se le han realizado propuestas de tesis, el personal de donde se debe seleccionar un tutor y los cotutores y el tema de la tesis a desarrollar.
2. El actor llena los datos requeridos y presiona el botón “Aceptar”.	2.1 El sistema verifica los datos y los almacena en la base de datos como una nueva propuesta de tesis. 2.1 El sistema envía un mensaje de confirmación y finaliza el caso de uso.
Flujos Alternativos de Eventos	
Acción del Actor	Respuesta del Sistema
	2.1 En caso de haber datos incorrectos o incompletos se muestra un mensaje de error indicando los campos incorrectos y se retorna al paso 2 finalizando el caso de uso.
Sección 2 “Ver detalles de la propuesta de tesis”	
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El actor selecciona una propuesta de tesis y presiona el botón “Ver detalles”.	1.1 El sistema muestra una interfaz con todos los datos de la propuesta de tesis seleccionada.
Sección 3 “Eliminar propuesta de tesis”	
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El actor presiona el botón “Eliminar propuesta de tesis”.	1.1 El sistema muestra un mensaje de diálogo para “Confirmar” o “Cancelar” la acción.
2. El actor oprime el botón “Confirmar”.	2.1 El sistema elimina toda la información referente a la propuesta de tesis, se retorna un mensaje informando que se ha eliminado y finaliza el caso de

	USO.
Flujos Alternativos de Eventos	
Acción del Actor	Respuesta del Sistema
2. El actor oprime el botón "Cancelar".	2.1 El sistema se mantiene en la misma interfaz y finaliza el caso de uso.
Sección 4 "Modificar datos de la propuesta de tesis"	
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El actor modifica los datos de la propuesta de tesis y presiona el botón "Enviar cambios".	1.1 El sistema verifica los datos y los guarda en la base de datos. 1.2 Muestra un mensaje de confirmación y finaliza en el caso de uso.
Flujos Alternativos de Eventos	
Acción del Actor	Respuesta del Sistema
1.2 Retorna al paso 1 y finaliza el caso de uso.	1.1 En caso de haber datos incorrectos o incompletos se envía un mensaje de error.

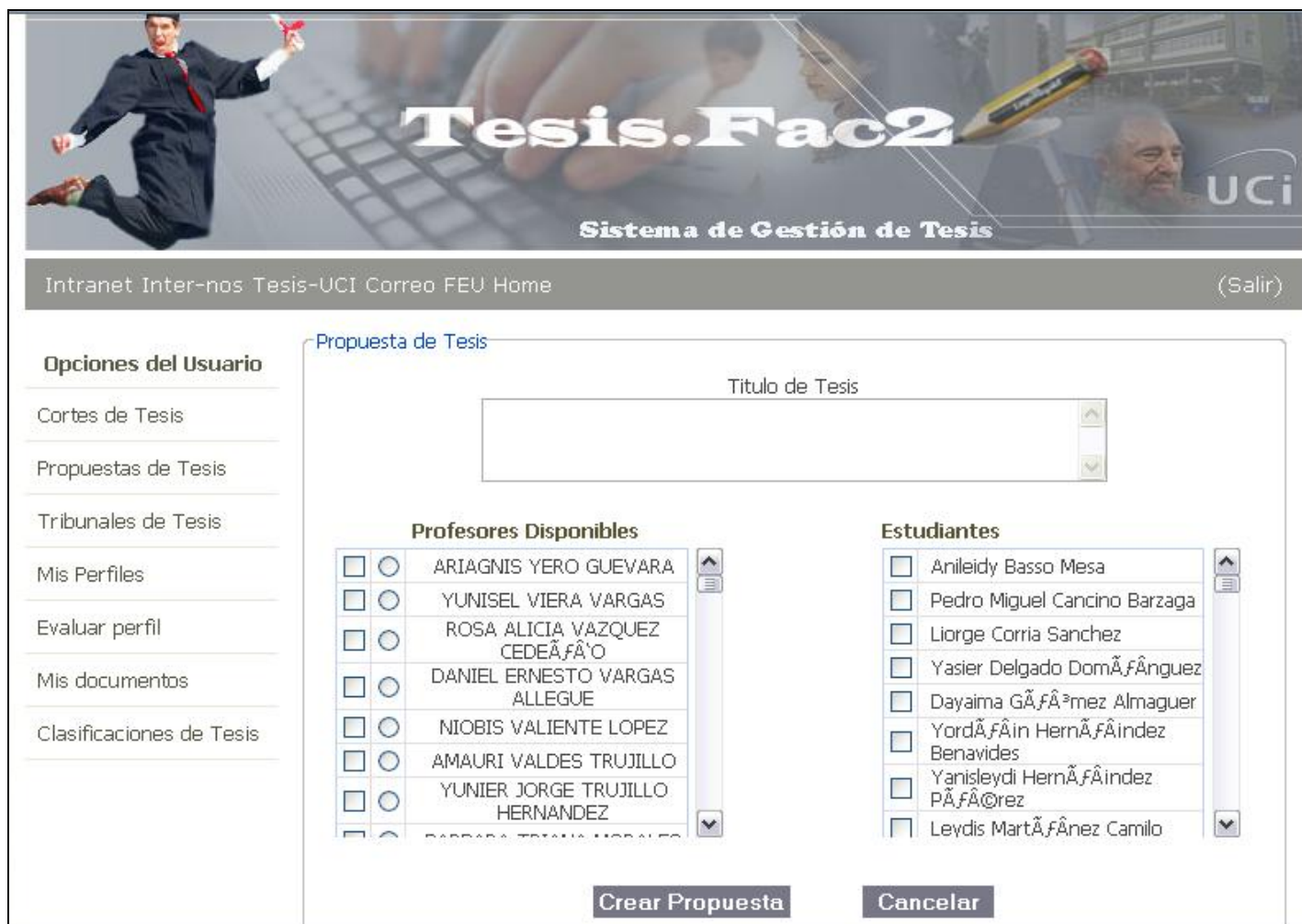


Tabla 19: Descripción del CUS “Gestionar propuestas de tesis”

Caso de Uso 8: “Gestionar notas de los cortes de tesis”.

Caso de Uso:	Gestionar notas de los cortes de tesis.
Actores:	Tribunal de tesis.
Resumen:	El caso de uso se inicia cuando el tribunal de tesis accede al sistema para registrar las notas que obtuvo cada tesis en los cortes realizados, así como en la pre defensa y defensa final, dando en cada momento la posibilidad de estimar la nota final de una tesis.

Precondiciones:	
Referencias:	RF 11
Prioridad:	Crítico
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El miembro del tribunal de tesis accede a la opción del menú "Evaluar tesis".	1.1 Muestra la interfaz correspondiente, enviando a la misma el listado de todas las tesis pertenecientes al tribunal en cuestión y brinda la opción de Ver detalles de las evaluaciones de los cortes de una tesis, así como agregar o modificar una nota.
2. El miembro del tribunal selecciona la acción a realizar. En caso de: 1. "Agregar nota" ir a la sección 1. 2. "Modificar nota" ir a la sección 2. 3. "Ver detalles de las evaluaciones" ir a sección 3"	
Sección 1 "Agregar nota"	
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
	1. El sistema muestra una interfaz con el listado de cortes indicando cuales han sido evaluados y brindando la opción de evaluar en caso de no tener ninguna nota asociada.
2. El miembro del tribunal presiona el botón evaluar correspondiente al corte que desee.	2.1 El sistema muestra una interfaz con las posibles evaluaciones a asignar.
3. El miembro del tribunal selecciona una evaluación y presiona el botón evaluar.	3.1 El sistema guarda la información recibida. 1.2 Muestra la misma interfaz con las evaluaciones y finaliza el caso de uso.
Flujos Alternativos de Eventos	
Acción del Actor	Respuesta del Sistema
2. El miembro del tribunal presiona el botón "Calcular nota final"	2.1 El sistema muestra un mensaje sobre la misma interfaz con la propuesta de nota final y finaliza el caso

	de uso.
Sección 2 “Modificar nota”	
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El miembro del tribunal presiona el botón “Modificar nota”.	1.1 El sistema muestra una interfaz con el listado de evaluaciones recibidas en cada corte de la tesis seleccionada y brinda la posibilidad de modificarlas.
2. El miembro del tribunal modifica la nota que estime conveniente y presiona el botón evaluar.	2.1 El sistema guarda los cambios y retorna a la interfaz anterior. 2.2 Muestra un mensaje de confirmación y finaliza el caso de uso.
Sección 3 “Ver detalles de las evaluaciones”	
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El miembro del tribunal presiona el botón “Ver detalles de las evaluaciones”.	1.1 El sistema muestra una interfaz con las evaluaciones recibidas en cada corte y finaliza el caso de uso.

Tabla 20: Descripción del CUS “Gestionar notas de los cortes de tesis”

Caso de Uso 9: “Gestionar cortes de tesis”.

Caso de Uso:	Gestionar cortes de tesis
Actores:	Comité de tesis
Resumen:	El caso de uso se inicia cuando un miembro del comité de tesis accede al sistema para publicar la información referente a los cortes de tesis, y que de esta forma esté disponible para los estudiantes y el personal involucrado.
Precondiciones:	
Referencias:	RF 6
Prioridad:	Crítico

Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El miembro del comité de tesis accede a la opción del menú "Cortes de tesis".	1.1 Muestra la interfaz correspondiente, enviando a la misma el listado de todos los cortes de tesis creados hasta el momento y brinda todas las posibles acciones a ejecutar.
2. El miembro del comité de tesis selecciona la acción a realizar. En caso de: 1. "Adicionar corte de tesis" ir a la sección 2. "Ver detalles" ir a la sección 2. 3. "Eliminar corte de tesis" ir a la sección 3. 4. "Modificar corte de tesis" ir a la sección 4.	



Intranet Inter-nos Tesis-UCI Correo FEU Home (Salir)

Opciones del Usuario

- Cortes de Tesis
- Propuestas de Tesis
- Tribunales de Tesis
- Mis Perfiles
- Evaluar perfil
- Mis documentos
- Clasificaciones de Tesis

Listado de cortes de tesis

Seleccione	Corte	Fecha	Tema a Evaluar	Acciones
<input type="checkbox"/>	1er Corte	2009-05-28	Metodologías y pruebas....	✖ 📄 🔄
<input type="checkbox"/>	2do Corte	2009-06-25	Todo lo que haya hecho.	✖ 📄 🔄

Eliminar todos
Crear Corte

Usted está registrado como Yadira Ruiz Constanten. Desde el IP: 10.7.8.30 Page rendered in 0.0912 seconds

Sección 1 “Adicionar corte de tesis”

Flujo Normal de Eventos

Acción del Actor	Respuesta del Sistema
1. El miembro del comité de tesis oprime el botón “Adicionar corte de tesis”.	1.1 El sistema muestra una interfaz con un formulario que contiene los campos a llenar.
2. El miembro del comité de tesis llena los datos requeridos y presiona el botón “Aceptar”.	2.1 El sistema verifica los datos y los almacena en la base de datos como un nuevo corte de tesis. 2.2 El sistema envía un mensaje de confirmación y finaliza el caso de uso.

Flujos Alternativos de Eventos

Acción del Actor	Respuesta del Sistema
	2.1 En caso de haber datos incorrectos o incompletos se muestra un mensaje de error indicando los campos incorrectos y se retorna al paso 2 finalizando el caso de uso.

Sección 2 “Ver detalles”

Flujo Normal de Eventos

Acción del Actor	Respuesta del Sistema
1. El miembro del comité de tesis selecciona un corte de tesis y presiona el botón “Ver detalles”.	1.1 El sistema muestra una interfaz con todos los datos del corte de tesis.

Sección 3 “Eliminar corte de tesis”	
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El miembro del comité de tesis presiona el botón “Eliminar corte de tesis”.	1.1 El sistema muestra un mensaje de diálogo para “Confirmar” o “Cancelar” la acción.
2. El miembro del comité de tesis oprime “Confirmar”.	2.1 El sistema elimina toda la información referente al corte de tesis, se retorna un mensaje informando que se ha eliminado y finaliza el caso de uso.
Flujos Alternativos de Eventos	
Acción del Actor	Respuesta del Sistema
2. El miembro del comité de tesis oprime “Cancelar”.	2.1 El sistema se mantiene en la misma interfaz y finaliza el caso de uso.
Sección 4 “Modificar datos del corte de tesis”	
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El miembro del comité de tesis modifica los datos del corte de tesis y presiona el botón “Enviar cambios”.	1.1 El sistema verifica los datos y los guarda en la base de datos. 1.2 Muestra un mensaje de confirmación y finaliza en el caso de uso.
Flujos Alternativos de Eventos	
Acción del Actor	Respuesta del Sistema
1.2 Retorna al paso 1 y finaliza el caso de uso.	1.1 En caso de haber datos incorrectos o incompletos se envía un mensaje de error.

Tabla 21: Descripción del CUS “Gestionar cortes de tesis”

Caso de Uso 10: “Gestionar usuarios”.

Caso de Uso:	Gestionar usuarios
Actores:	Administrador
Resumen:	El caso de uso se inicia cuando el sistema carga automáticamente los

	datos de los usuarios predeterminados y luego el administrador accede para modificar sus datos o agregar algunos nuevos. De esta forma se logra controlar las acciones que cada uno de ellos puede realizar.	
Precondiciones:	El sistema debe haber cargado los datos requeridos de los estudiantes de 5to año y del claustro de profesores de la Facultad 2.	
Referencias:	RF 10	
Prioridad:	Crítico	
Flujo Normal de Eventos		
	Acción del Actor	Respuesta del Sistema
	1. El administrador accede a la opción del menú "Usuarios del sistema".	1.1 Muestra la interfaz correspondiente, enviando a la misma el listado de todos los usuarios del sistema creados hasta el momento y brinda todas las posibles acciones a ejecutar.
	2. El administrador selecciona la acción a realizar. En caso de: <ul style="list-style-type: none"> 1. "Adicionar nuevo usuario" ir a la sección 1. 2. "Ver detalles" ir a la sección 2. 3. "Eliminar usuario" ir a la sección 3. 4. "Modificar datos del usuario" ir a la sección 4. 	

//10.31.17.231:5800/CodeIgniter_1.7.1X/index.php/user_controller/paginar_usuario

Sistema de Gestión de Tesis

Intranet Inter-nos Tesis-UCI Correo FEU Home (Salir)

Opciones del Usuario

- Proyectos
- Roles de usuarios
- Usuarios
- Areas

Listado de usuario

Seleccione	Nombre	Apellidos	Usuario	Nombre del rol	Acciones
<input type="checkbox"/>	ABDEL EDUARDO	MARTINEZ VALVERDE		Personal	
<input type="checkbox"/>	Abel	Olivares Castellanos		Estudiante	
<input type="checkbox"/>	Abel Alejandro	Yáñez Moncada		Estudiante	
<input type="checkbox"/>	Administrador	admin admin	Administrador	Administrador	
<input type="checkbox"/>	ADONIS	RODRIGUEZ FERNANDEZ		Personal	
<input type="checkbox"/>	ADRIAN	MARANJE AGRAMONTE		Personal	
<input type="checkbox"/>	Adrian	Páez Mejias		Estudiante	
<input type="checkbox"/>	Adrian Roberto	Correa Borrego		Estudiante	
<input type="checkbox"/>	AILIN	ALARCON FERRA		Personal	
<input type="checkbox"/>	AIMEL	CASTELLÁN BELTRÁN		Personal	

1 2 3 > Ultimo >

Eliminar todos **Nuevo usuario**

Sección 1 “Adicionar nuevo usuario”

Flujo Normal de Eventos

Acción del Actor	Respuesta del Sistema
1. El administrador oprime el botón “Adicionar usuario”.	1.1 El sistema muestra una interfaz con un formulario que contiene los campos a llenar.
2. El administrador llena los datos requeridos y presiona el botón “Aceptar”.	2.1 El sistema verifica los datos y los almacena en la base de datos como un nuevo usuario. 2.2 El sistema envía un mensaje de confirmación y finaliza el caso de uso.



Opciones del Usuario

- Proyectos
- Roles de usuarios
- Usuarios
- Areas

Crear Usuario

<p>Nombre</p> <input type="text"/>	<p>Areas</p> <input type="text"/>
<p>Apellidos</p> <input type="text"/>	<p>Roles</p> <ul style="list-style-type: none"> <input type="radio"/> Administrador <input type="radio"/> Estudiante <input type="radio"/> Personal <input type="radio"/> Jefe Tribunal <input type="radio"/> Comite de tesis
<p>Usuario</p> <input type="text"/>	
<p>Solapin</p> <input type="text"/>	

Flujos Alternativos de Eventos

Acción del Actor	Respuesta del Sistema
	2.1 En caso de haber datos incorrectos o incompletos se muestra un mensaje de error indicando los campos incorrectos y se retorna al paso 2 finalizando el caso de uso.

Sección 2 "Ver detalles"

Flujo Normal de Eventos

Acción del Actor	Respuesta del Sistema
1. El administrador selecciona un usuario y presiona el botón "Ver detalles".	1.1 El sistema muestra una interfaz con todos los datos del usuario seleccionado.

Sección 3 “Eliminar usuario”	
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El administrador presiona el botón “Eliminar usuario”.	1.1 El sistema muestra un mensaje de diálogo para “Confirmar” o “Cancelar” la acción.
2. El administrador oprime “Confirmar”.	2.1 El sistema elimina toda la información referente al usuario, se retorna un mensaje informando que se ha eliminado y finaliza el caso de uso.
Flujos Alternativos de Eventos	
Acción del Actor	Respuesta del Sistema
2. El administrador oprime “Cancelar”.	2.1 El sistema se mantiene en la misma interfaz y finaliza el caso de uso.
Sección 4 “Modificar datos del usuario”	
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El administrador modifica los datos del usuario y presiona el botón “Enviar cambios”.	1.1 El sistema verifica los datos y los guarda en la base de datos. 1.2 Muestra un mensaje de confirmación y finaliza en el caso de uso.
Flujos Alternativos de Eventos	
Acción del Actor	Respuesta del Sistema
1.2 Retorna al paso 1 y finaliza el caso de uso.	1.1 En caso de haber datos incorrectos o incompletos se envía un mensaje de error.

Tabla 22: Descripción del CUS “Gestionar usuarios”

Caso de Uso 11: “Realizar comentarios”.

Caso de Uso:	Realizar comentarios
Actores:	Miembros del comité de tesis, miembros de los tribunales de tesis, líderes de proyecto y tutores.
Resumen:	El caso de uso se inicia cuando algún personal que se encuentre revisando alguno

	de los documentos, ya sea perfil o el propio documento de tesis agregue algún comentario para argumentar alguna decisión o para hacer alguna sugerencia o señalamiento.
Precondiciones:	
Referencias:	RF 12
Prioridad:	Opcional
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El actor presiona el botón “Agregar comentario”.	1.1 Muestra una interfaz con un cuadro de texto habilitado para la escritura.
2. El actor escribe su comentario y presiona el botón “Enviar”.	2.1 El sistema almacena el comentario en la base de datos y finaliza el caso de uso.

Tabla 23: Descripción del CUS “Realizar comentarios”

Caso de Uso 12: “Realizar reportes”.

Caso de Uso:	Realizar reportes
Actores:	Comité de tesis
Resumen:	El caso de uso se inicia cuando un miembro del comité de tesis accede al sistema para obtener reportes de acuerdo a los criterios seleccionados.
Precondiciones:	
Referencias:	RF 13
Prioridad:	Secundario
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El miembro del comité de tesis accede a la opción del menú “Reportes”.	1.1 Muestra la interfaz correspondiente, enviando a la misma los posibles criterios para obtener reportes.
2. El miembro del comité de tesis selecciona un criterio y presiona el botón “Aceptar”.	2.1 Muestra los resultados del reporte seleccionado y finaliza el caso de uso.

Tabla 24: Descripción del CUS “Realizar reportes”

Caso de Uso 13: “Consultar comentarios”.

Caso de Uso:	Consultar comentarios	
Actores:	Estudiante, Tutor	
Resumen:	El caso de uso se inicia cuando el estudiante revisa los comentarios realizados a sus documentos.	
Precondiciones:		
Referencias:	RF 12	
Prioridad:	Opcional	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Sistema	
1. El actor presiona el botón “Comentarios”.	1.1 El sistema muestra una interfaz con los comentarios recibidos, brindando la posibilidad de ver los detalles de cada comentario o eliminar alguno en caso que así lo desee.	
2. En caso de: 1. “Ver detalles” ir a la sección 1. 2. “Eliminar comentario” ir a la sección 2.		
Sección 1 “Ver detalles”		
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Sistema	
1. El actor selecciona un comentario y presiona el botón “Ver comentario”.	1.1 El sistema muestra una interfaz con el contenido del comentario y finaliza el caso de uso.	
Sección 2 “Eliminar comentario”		
Acción del Actor	Respuesta del Sistema	
1. El actor presiona el botón “Eliminar comentario”.	1.1 El sistema muestra un mensaje para confirmar la acción.	

2. El estudiante presiona el botón “Confirmar”.	2.1 El sistema elimina el comentario y finaliza el caso de uso.
Flujos Alternativos de Eventos	
Acción del Actor	Respuesta del Sistema
2. El actor presiona el botón “Cancelar”.	2.1 El sistema mantiene la misma interfaz y finaliza el caso de uso.

Tabla 25: Descripción del CUS “Consultar comentarios”

Caso de Uso 14: “Notificar información del perfil de tesis”.

Caso de Uso:	Notificar información del perfil de tesis	
Actores:	Estudiante, Tutor, Miembro del tribunal, Miembro comité de tesis, Líder de proyecto	
Resumen:	El caso de uso se inicia cuando alguno de los actores señalados accede al sistema para obtener información de los perfiles de tesis a lo que tiene acceso y en caso de estimarlo conveniente agregar algún comentario.	
Precondiciones:		
Referencias:	RF 7	
Prioridad:	Crítico	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Sistema	
1. El actor presiona la opción del menú “Perfil de tesis”	1.1 El sistema muestra una interfaz de acuerdo a los permisos con que cuente el usuario autenticado. 1. En caso de ser estudiante muestra una interfaz con toda la información de su perfil de tesis y finaliza el caso de uso. 2. En otro caso ir a sección 1.	
Sección 1		
Acción del Actor	Respuesta del Sistema	
	1. El sistema muestra una interfaz con el listado de los perfiles de las tesis a las que tiene acceso brindando la posibilidad de ver sus detalles.	

2. El actor selecciona un perfil de tesis y presiona el botón “Ver detalles del perfil”.	2.1 Muestra una interfaz con la información del perfil escogido y finaliza el caso de uso.
--	--

Tabla 26: Descripción del CUS “Notificar información del perfil de tesis”

Caso de Uso 15: “Revisar documento de tesis”.

Caso de Uso:	Revisar documento de tesis.	
Actores:	Tutor, Miembro del tribunal, Miembro comité de tesis, Líder de proyecto	
Resumen:	El caso de uso se inicia cuando alguno de los actores señalados accede al sistema para hacer una revisión del documento de tesis, de modo que el estudiante conozca los señalamientos realizados a su trabajo y en caso de estimarlo conveniente, realizar algún comentario por parte del personal involucrado en la tesis.	
Precondiciones:		
Referencias:	RF 8	
Prioridad:	Crítico	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Sistema	
1. El actor accede a la opción del menú “Documentos de tesis”.	1.1 El sistema muestra una interfaz con los documentos de las tesis a las que tiene acceso de acuerdo a su rol.	
2. El actor selecciona un documento de tesis.	2.1 El sistema brinda las opciones de “Descargar documento” y de “Agregar comentario”.	
3. En caso de: 1. “Descargar documento” ir a la sección 1. 2. “Agregar comentario” ir a la descripción del caso de uso 12.		
Sección 1 “Descargar documento de tesis”		
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Sistema	
1 El actor oprime el botón “Bajar”.	1.1 El sistema muestra un “File Download” y brinda las opciones de “Abrir”, “Guardar” y “Cancelar”.	

2 El actor oprime el botón "Guardar".	2.1 El sistema muestra un "Save Dialog" para especificar la ruta donde se descargará el documento dando las opciones de "Guardar" y "Cancelar".
3. El actor especifica la ruta y oprime "Guardar".	2.3 El sistema realiza la descarga y finaliza el caso de uso.
Flujos Alternativos de Eventos	
Acción del Actor	Respuesta del Sistema
2. El actor oprime "Abrir".	2.1 El sistema abre el documento visualizando todo su contenido y formato finalizando el caso de uso.
2. El actor oprime "Cancelar".	2.1 a) El sistema desactiva el "File Download" y retorna a la interfaz finalizando el caso de uso.

Tabla 27: Descripción del CUS "Revisar documento de tesis"

Caso de Uso 16: "Eliminar documento de tesis".

Caso de Uso:	Revisar documento de tesis.
Actores:	Estudiante
Resumen:	El caso de uso se inicia cuando un estudiante accede al sistema para eliminar un documento de tesis.
Precondiciones:	La sesión iniciada debe ser la de un estudiante
Referencias:	RF 8
Prioridad:	Crítico
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El estudiante oprime el botón "Eliminar documento".	1.1 El sistema muestra un mensaje para confirmar que desea ejecutar la acción.
2. El estudiante presiona el botón Aceptar.	2.1 El sistema elimina los datos del documento seleccionado y finaliza el caso de uso.
Flujos Alternativos de Eventos	
Acción del Actor	Respuesta del Sistema

3. El estudiante oprime "Cancelar".	3.1 El sistema retorna a la interfaz anterior y finaliza el caso de uso.

Tabla 28: Descripción del CUS "Eliminar documento de tesis"

Anexo 5. Diagramas de interacción del diseño

Caso de Uso “Autenticar usuario”.

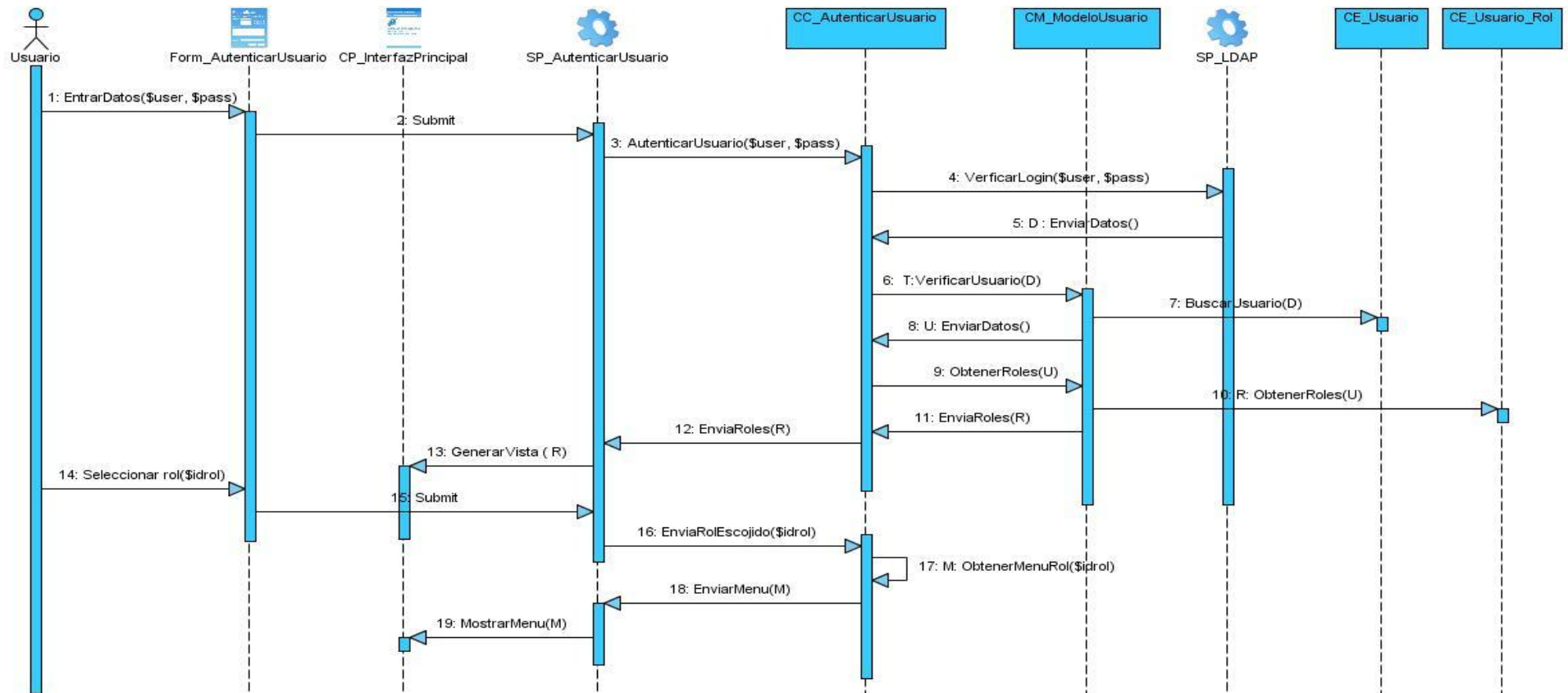


Tabla 29: Diagrama de secuencia del CUS “Autenticar usuario”

Caso de Uso “Evaluar perfil”

Sección “Aceptar perfil”

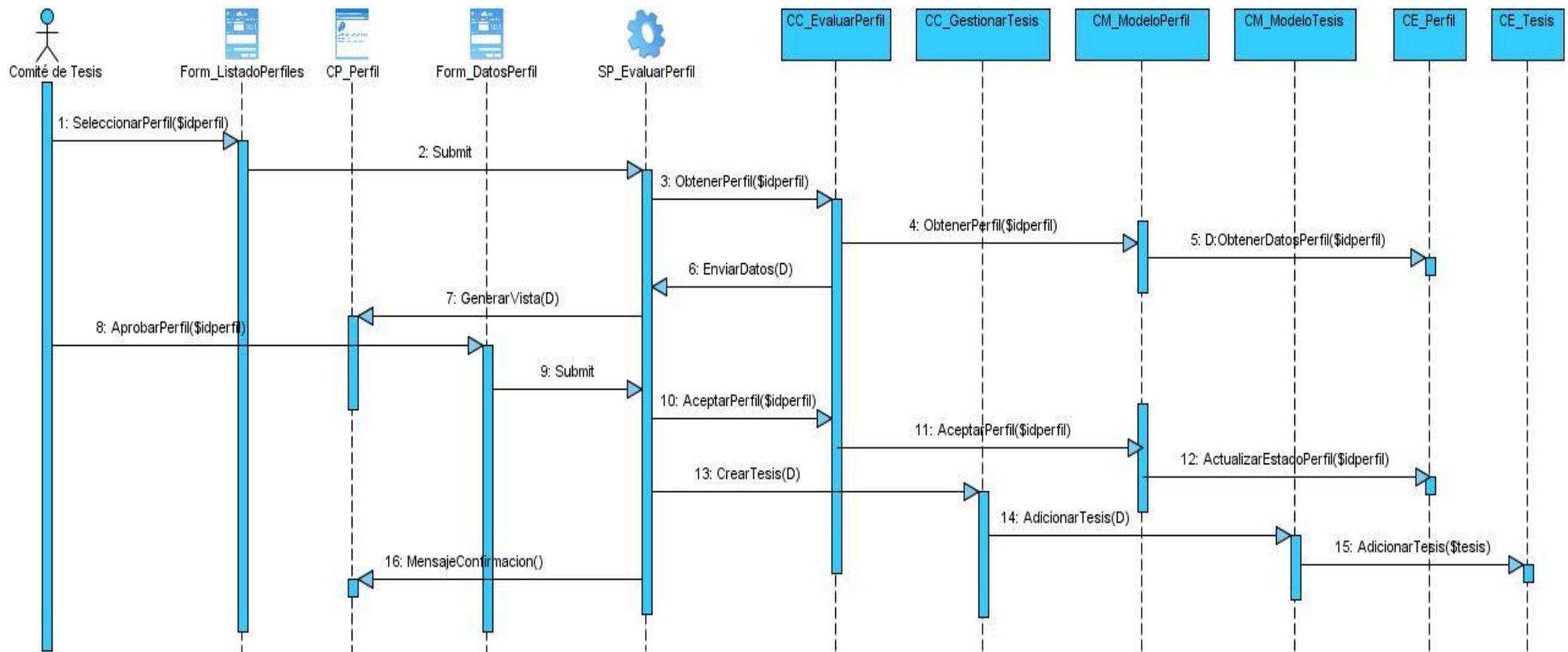


Tabla 30: Diagrama de secuencia del CUS “Evaluar perfil”. Sección “Aceptar perfil”

Caso de Uso "Evaluar perfil"

Sección "Rechazar perfil"

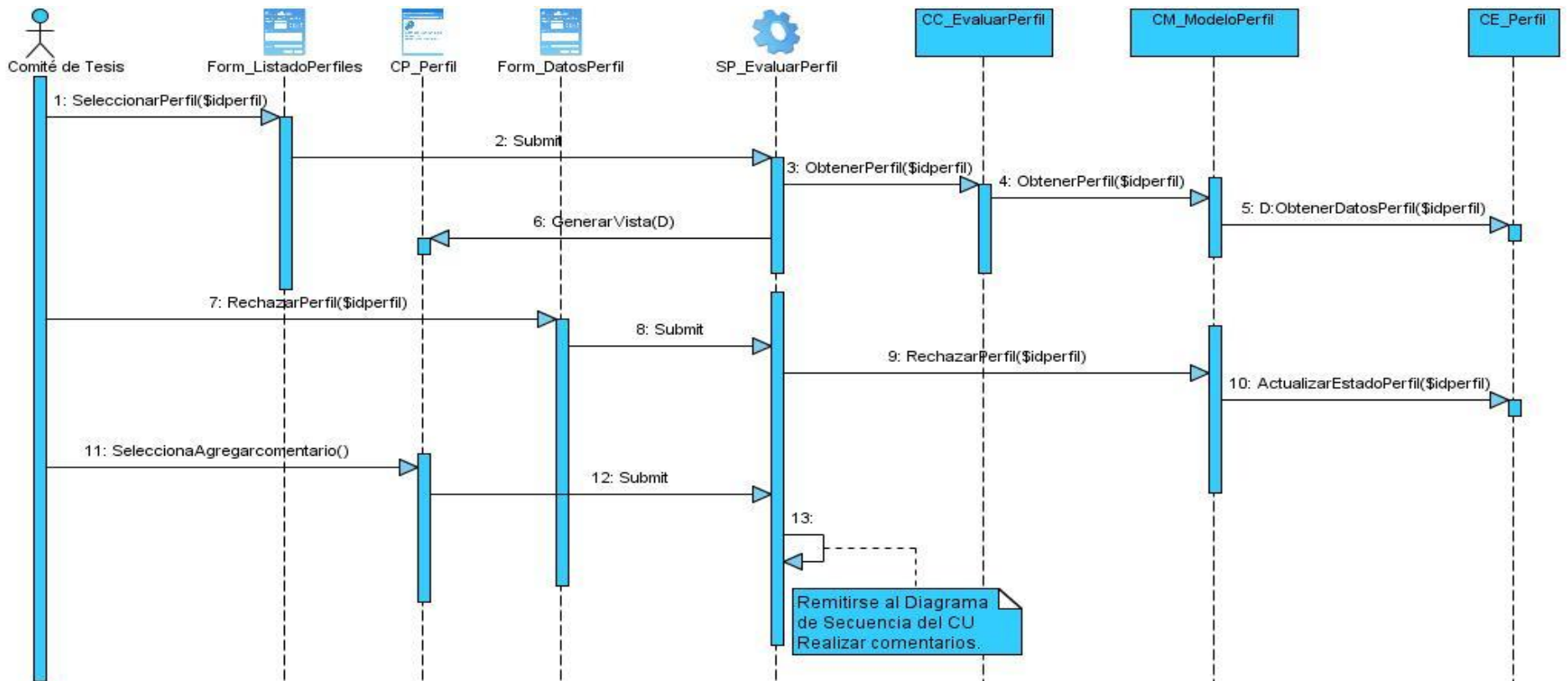


Tabla 31: Diagrama de secuencia del CUS "Evaluar perfil". Sección "Rechazar perfil"

Caso de Uso "Gestionar cortes de tesis"

Sección "Eliminar corte de tesis"

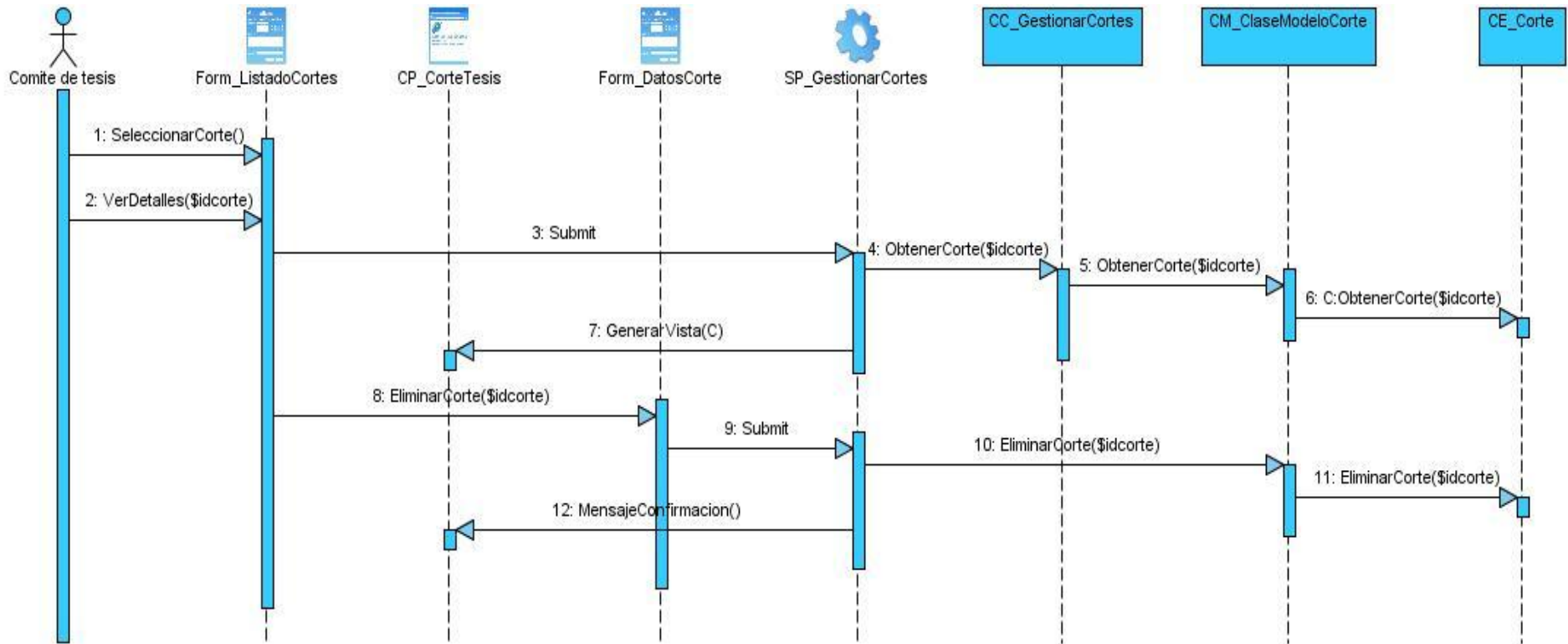


Tabla 32: Diagrama de secuencia del CUS "Gestionar corte de tesis". Sección "Eliminar corte"

Caso de Uso “Gestionar documento de tesis”

Sección “Cargar documento de tesis”

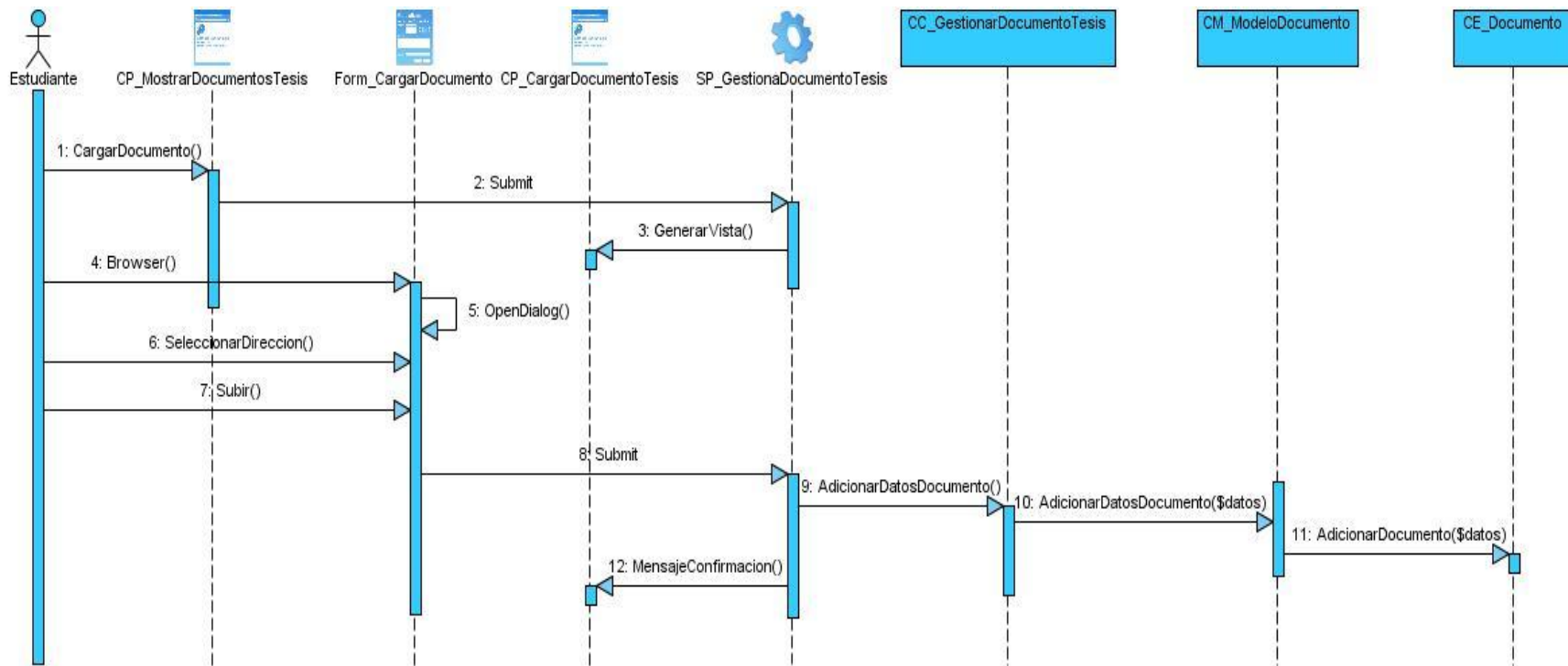


Tabla 33: Diagrama de secuencia del CUS “Gestionar documento de tesis”. Sección “Cargar documento”

Caso de Uso “Gestionar perfil de tesis”

Sección “Crear perfil de tesis usando propuesta”

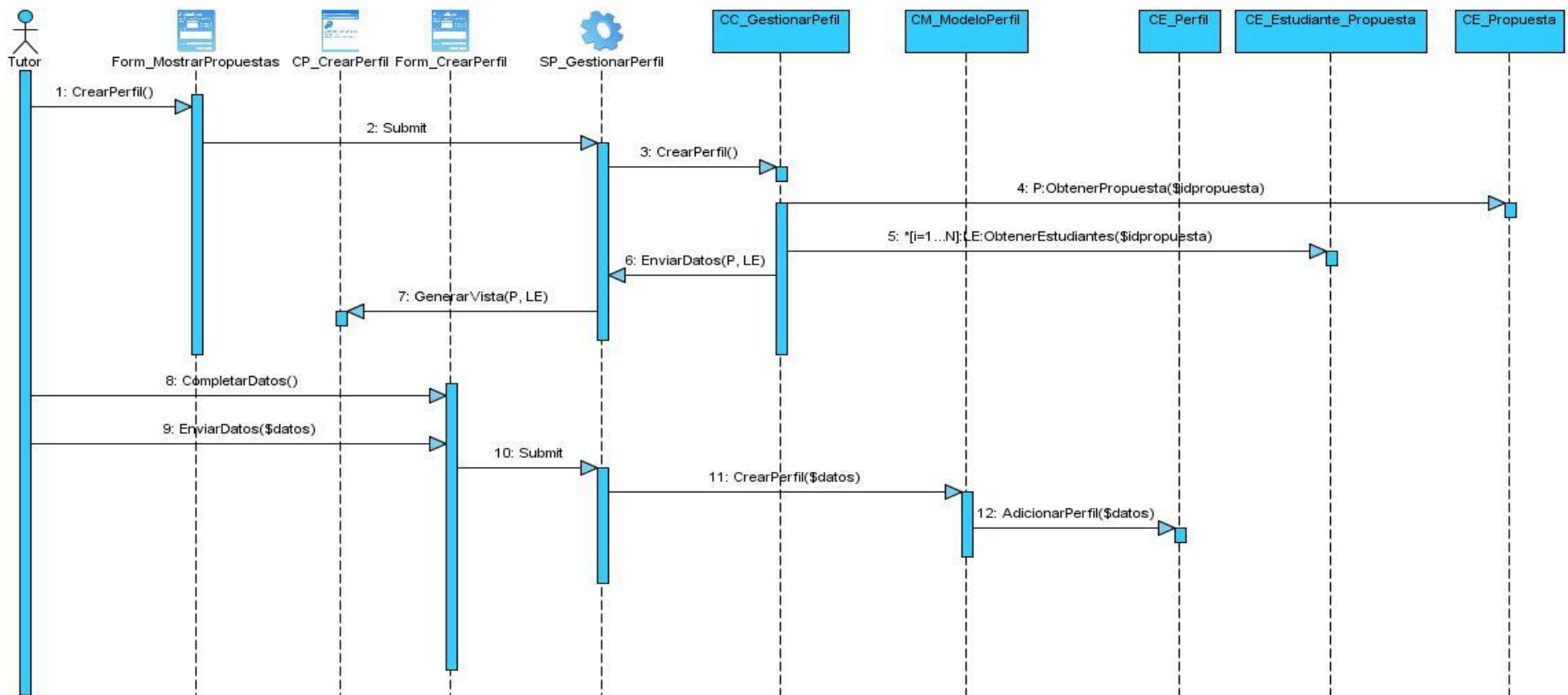


Tabla 34: Diagrama de secuencia del CUS “Gestionar perfil de tesis”. Sección “Crear perfil usando propuesta”

Sección "Mostrar perfil de tesis"

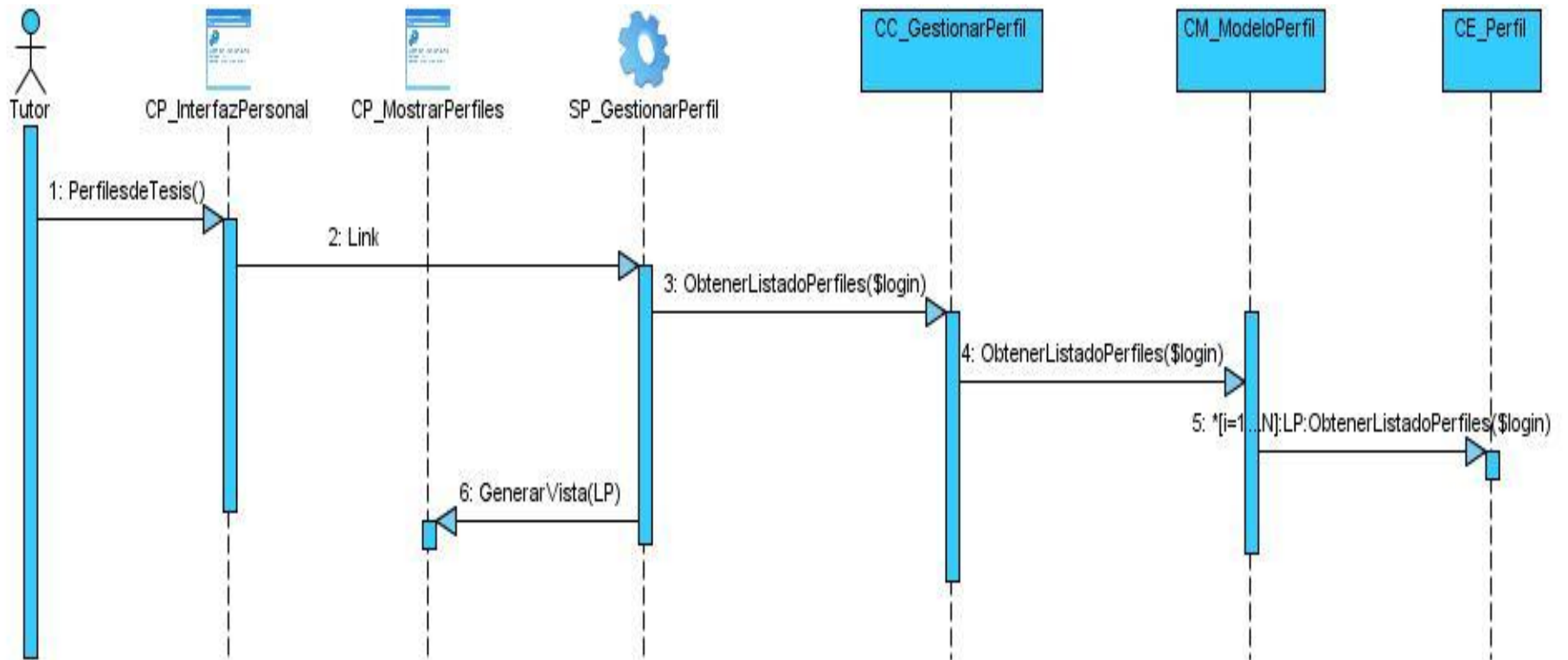


Tabla 35: Diagrama de secuencia del CUS "Gestionar perfil de tesis". Sección "Mostrar perfil"

Caso de Uso “Realizar comentario”

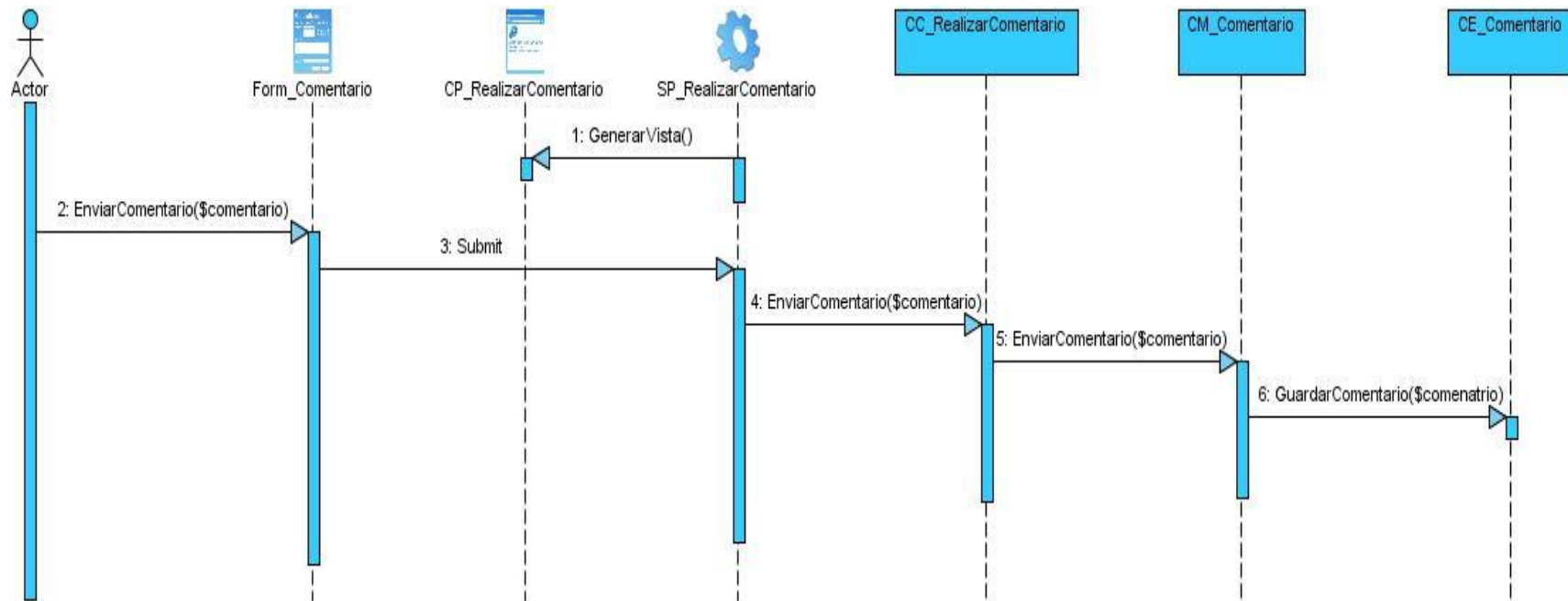


Tabla 36: Diagrama de secuencia del CUS “Realizar comentario”

Glosario de Términos

- **API**, una interfaz de programación de aplicaciones o API (del inglés Application Programming Interface) es el conjunto de funciones y procedimientos que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción.
- **AWK**, es un lenguaje de programación diseñado para procesar datos basados en texto, ya sean ficheros o flujos de datos.
- **CGI (Common Gateway Interface)**, es una importante tecnología de la World Wide Web que permite a un cliente (explorador web) solicitar datos de un programa ejecutado en un servidor web. CGI especifica un estándar para transferir datos entre el cliente y el programa.
- **DDL (Data Definition Lenguaje - lenguaje de definición de datos)**: las sentencias DDL son aquellas utilizadas para la creación de una base de datos y todos sus componentes: tablas, índices, relaciones, disparadores (triggers), procedimientos almacenados, entre otros.
- **DML (Data Manipulation Lenguaje - lenguaje de manipulación de datos)**: las sentencias DML son aquellas utilizadas para insertar, borrar, modificar y consultar los datos de una base de datos.
- **HTML**, siglas de HyperText Markup Language (Lenguaje de Marcas de Hipertexto), es el lenguaje de marcado predominante para la construcción de páginas web.
- **HTTP (HyperText Transfer Protocol - Protocolo de transferencia de hipertexto)**, es el protocolo usado en cada transacción de la Web. Define la sintaxis y la semántica que utilizan los elementos software de la arquitectura web (clientes, servidores, proxies) para comunicarse. Es un protocolo orientado a transacciones y sigue el esquema petición-respuesta entre un cliente y un servidor.
- **HTTPS (Hypertext Transfer Protocol Secure - Protocolo seguro de transferencia de hipertexto)**, es un protocolo de red basado en el protocolo HTTP, destinado a la transferencia segura de datos de hipertexto, es decir, es la versión segura de HTTP.

- **IIS (Internet Information Server)**, es una serie de servicios para los ordenadores que funcionan con Windows. Proporciona las herramientas y funciones necesarias para administrar de forma sencilla un servidor Web seguro.
- **JSP (Java Sever pages)**, es una tecnología orientada a crear páginas web con programación en Java.
- **LDAP (Lightweight Directory Access Protocol - Protocolo Ligero de Acceso a Directorios)**, es un protocolo a nivel de aplicación que permite el acceso a un servicio de directorio ordenado y distribuido para buscar diversa información en un entorno de red. LDAP también es considerado una base de datos optimizada para las operaciones de lectura y búsqueda, a la que pueden realizarse consultas.
- **Licencia GPL (General Public Licence - Licencia Pública General)** es una licencia orientada principalmente a proteger la libre distribución, modificación y uso de software. Su propósito es declarar que el software cubierto por esta licencia es software libre y protegerlo de intentos de apropiación que restrinjan esas libertades a los usuarios.
- **Lisp**, es el segundo lenguaje de programación más antiguo (después de Fortran) de alto nivel, usado generalmente en el mundo de la inteligencia artificial.
- **Multihilo**, significa que puede procesar varias peticiones a la misma vez e independiente de las demás.
- **Multiusuario**, significa que soporta el acceso concurrente de varios usuarios.
- **Sed (stream editor - editor de flujos)**, es un editor de texto que acepta como entrada un flujo (que puede ser la entrada estándar o un archivo), lo lee y modifica línea a línea mostrando el resultado en la salida estándar o pantalla.
- **Servlets**, generar páginas web de forma dinámica a partir de los parámetros de la petición que envíe el navegador web.

- **SQL (Structured Query Language – Lenguaje de consulta estructurado)**, es un lenguaje declarativo de acceso a bases de datos.
- **TCP/IP (Transmission-Control-Protocol/Internet Protocol - Protocolo de Control de Transmisión/ Protocolo de Internet)**, un sistema de protocolos que hacen posibles servicios Telnet, FTP, E-mail, y otros entre ordenadores que no pertenecen a la misma red.
- **UML (Unified Modeling Language – Lenguaje Unificado de Modelado)**, es un lenguaje de modelado visual que se usa para especificar, visualizar, construir y documentar artefactos de un sistema de software.
- **URL (Uniform Resource Locator – localizador uniforme de recursos)**, es una secuencia de caracteres, de acuerdo a un formato estándar, que se usa para nombrar recursos por su localización.
- **XHTML**, acrónimo inglés de eXtensible Hypertext Markup Language (lenguaje extensible de marcado de hipertexto), es el lenguaje de marcado pensado para sustituir a HTML como estándar para las páginas web.
- **XML (Extensible Markup Language)** es un metalenguaje que permite definir la gramática de lenguajes específicos. Es una manera de definir lenguajes para diferentes necesidades.