



Facultad 8

Aplicación Web para el control del Proceso de Tesis de Grado
en la Facultad 8.

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

Autores:

Reynaldo Heredia Rodríguez.

Emilio Salcerio Paz.

Tutor:

Yasim Mirabal Vetano.

Ciudad de la Habana

2008, Año 50 de la Revolución.

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 firmo la presente a los ____ días del mes de _____ del año _____.

Firma del Autor.

Reynaldo Heredia Rodríguez.

Firma del Autor.

Emilio Salcerio Paz.

Firma del Tutor.

Ing. Yasim Mirabal Vetano.

El éxito de los hombres no se mide por su éxito inmediato, sino por su éxito definitivo, no se mide por el dinero que acumularon, sino por el resultado de sus obras.

José Martí.

AGRADECIMIENTOS:

*A mis padres **Maricel Rodríguez Fernández** y **Reynaldo Heredia Lavadie**, y a mi hermano*

***Reinier Heredia Rodríguez** por el apoyo, el ejemplo y por saberme guiar siempre por el camino correcto.*

*A mi abuelo **Francisco Rodríguez Basulto** por haberme apoyado y darme fuerzas para continuar, yo se que estaría muy orgulloso de mi al verme graduado.*

*A mi novia **Aida Sacerio Martínez** (Aidita) por su comprensión y amor.*

*A mis amigos y hermanos **Alejandro Valdés Villarruvia**, **Aramis Rescalla Pupo** y **Dairon Pérez Roldan** que le debo mucho en mi carrera y en la vida.*

*A mi tutor **Yasim Mirabal Vetano**, por ayudarme siempre que lo necesitaba.*

*A mis **compañeros de estudio** por toda la ayuda que me han prestado.*

A todo el que de una forma u otra ha hecho posible la realización de este sueño.

Reynaldo Heredia Rodríguez.

*A mis **padres** por haberme inspirado confianza en todo momento y por haberme guiado siempre por el camino correcto.*

*A mis **hermanos** por haber confiado en mí aun en los momentos más difíciles de mi carrera.*

*A mi novia **Danisleydi Noroña Blanco** (Dani), por apoyarme y soportarme en todo momento.*

*A todos mis **primos** por ayudarme a poder decidir cuando estaba confundido.*

*A mis **amigos** que siempre han estado en los momentos que los he necesitado, **Deiler Sevilla Fernández, Alexeider Rodríguez Rojas, Yasser Chacón Cabrera, Yanirys Silva Lemus.***

*A **José Sánchez de la Peña**, un amigo más al cual le debo muchas de las cosas que se de Drupal y además por sugerirme el título que lleva la aplicación.*

*A mi tutor **Yasim Mirabal Vetano**, por ayudarme siempre que lo necesitaba.*

A todo el que de una forma u otra ha hecho posible que terminara mis estudios bien.

Emilio Salcerio Paz.

*A nuestro Comandante **Fidel Castro Ruz** a nuestro presidente **Raúl Castro Ruz** y a la Revolución Cubana por darnos la oportunidad de convertirnos en profesionales.*

Muchas Gracias.

DEDICATORIA:

A nuestros padres, hermanos, familiares y amigos...

RESUMEN

En este trabajo se presenta un una aplicación Web dinámica con el objetivo de optimizar el trabajo y la documentación referente al Proceso de Tesis de Grado en la Facultad 8. Esta aplicación es independiente del sistema operativo donde se ejecute y presenta un requerimiento de hardware mínimo, brinda una amplia información sobre todo el tema referente a las tesis de grado, minimiza el papeleo y agiliza el flujo de información ahorrando así tiempo y recursos a la facultad 8. El sistema cuenta con una interfaz Web amigable y fácil de usar permitiendo así que un mayor número de usuarios puedan disfrutar de sus servicios.

ÍNDICE DE CONTENIDO

INTRODUCCIÓN	1
Capitulo I. Fundamentación teórica	3
1.1 Introducción	3
1.2 Estado del arte	3
1.3 Propuesta de diseño	4
1.3.1 CMS (Content Management System)	4
CMS seleccionado (Drupal).....	6
1.3.2 Lenguajes de programación Web	7
JAVA	8
PERL.....	8
PHP	10
Lenguaje seleccionado (PHP)	10
1.3.3 Servidores Web	11
Apache	11
Lighttpd.....	12
IIS	13
Thttpd	13
Servidor Web seleccionado (Apache).....	14
1.3.4 Gestor de base de datos	15
MySQL	15
PostgreSQL.....	16
Firebird	16
Servidor de Base de datos seleccionado (MySQL)	16
1.3.5 Metodologías	17
RUP.....	17
XP.....	19
Metodología seleccionada (XP)	19
1.3.6 Plataforma	21

Aplicación Web.....	21
Conclusiones	22
Capitulo II. Características del Sistema	24
2.1 Introducción.....	24
2.2. Objeto de estudio.....	24
2.2.1 Situación Problémica	24
2.2.2 Problema.....	24
2.2.3 Objeto de Automatización	24
2.2.3 Información que se maneja	25
2.2.4 Propuesta del Sistema	25
2.4.5 Arquitectura del sistema propuesto.....	25
2.4.6 Interfaz de usuario.....	26
2.4.7 Requisitos no funcionales del sistema.....	28
Requerimientos de apariencia o interfaz externa	28
Requerimientos de usabilidad	28
Requerimientos de rendimiento	28
Requerimiento de portabilidad	29
Requerimientos de seguridad	29
Requerimientos de software.....	29
Requerimientos de hardware	29
Restricción en el diseño y la implementación	29
Conclusiones.....	30
Capitulo III. Exploración y planificación	54
3.1 Introducción.....	54
3.2 Fase de exploración	54
3.2.1 Historias de Usuario.....	54
3.3 Planificación	71
3.3.1. Estimación de esfuerzo por Historias de Usuario.....	71
3.3.2. Plan de Iteraciones	72
3.3.3. Iteración 1	72
3.3.4. Iteración 2	72
3.3.5. Iteración 3	72

3.3.6. Iteración 4	73
3.3.6 Plan de duración de las iteraciones	73
3.3.7 Plan de entregas	74
Conclusiones.....	53
Capitulo IV. Implementación y Pruebas.....	74
4.1 Introducción	74
4.2 Iteración 1	75
4.3 Iteración 2	81
4.4 Iteración 3	86
4.4 Iteración 4	96
4.6 Pruebas	103
4.7 Pruebas de Aceptación.....	104
4.8 Diagramas de clases	104
Conclusiones.....	104
Capitulo V. Estudio de Factibilidad	87
5.1 Introducción	87
5.2 Características del proyecto.....	87
5.3 Cálculo de instrucciones fuentes, esfuerzo, tiempo de desarrollo, cantidad de hombres y costo	89
Conclusiones.....	92
Conclusiones generales	93
Recomendaciones.....	94
Bibliografía	95
Anexo I Diagramas de clases	98
Anexo II Pruebas de Aceptación.....	102
Anexo III Opiniones del Usuario y del Tutor con respecto al trabajo	115
Glosario de términos.....	117

ÍNDICE DE TABLAS

TABLA 3.2. 1 HISTORIA DE USUARIO AUTENTICAR USUARIO.....	54
TABLA 3.2. 2 HISTORIA DE USUARIO ELIMINAR USUARIO.....	55
TABLA 3.2. 3 HISTORIA DE USUARIO CREAR DE COMITÉ DE TESIS.....	56
TABLA 3.2. 4 HISTORIA DE USUARIO MODIFICAR DE COMITÉ DE TESIS.....	56
TABLA 3.2. 5 HISTORIA DE USUARIO ELIMINAR DE COMITÉ DE TESIS.....	57
TABLA 3.2. 6 HISTORIA DE USUARIO CREAR TRIBUNAL.....	57
TABLA 3.2. 7 HISTORIA DE USUARIO MODIFICAR TRIBUNAL.....	58
TABLA 3.2. 8 HISTORIA DE USUARIO ELIMINAR TRIBUNAL.....	59
TABLA 3.2. 9 HISTORIA DE USUARIO PUBLICAR TEMA DE TESIS.....	60
TABLA 3.2. 10 HISTORIA DE USUARIO ELIMINAR TEMA DE TESIS.....	60
TABLA 3.2. 11 HISTORIA DE USUARIO ELIMINAR TEMA DE TESIS.....	61
TABLA 3.2. 12 HISTORIA DE USUARIO MODIFICAR TEMA DE TESIS.....	62
TABLA 3.2. 13 HISTORIA DE USUARIO PUBLICAR CRONOGRAMA.....	63
TABLA 3.2. 14 HISTORIA DE USUARIO ASIGNAR TEMA DE TESIS.....	63
TABLA 3.2. 15 HISTORIA DE USUARIO EVALUAR TEMA DE TESIS.....	64
TABLA 3.2. 16 HISTORIA DE USUARIO LISTAR PROFESORES.....	65
TABLA 3.2. 17 HISTORIA DE USUARIO LISTAR GRUPOS DE QUINTO AÑO.....	65
TABLA 3.2. 18 HISTORIA DE USUARIO LISTAR ESTUDIANTES SIN TESIS.....	66
TABLA 3.2. 19 HISTORIA DE USUARIO ADICIONAR ESTUDIANTES.....	67
TABLA 3.2. 20 HISTORIA DE USUARIO ELIMINAR ESTUDIANTES.....	67
TABLA 3.2. 21 HISTORIA DE USUARIO ADICIONAR PROFESORES.....	68
TABLA 3.2. 22 HISTORIA DE USUARIO ELIMINAR PROFESORES.....	69
TABLA 3.2. 23 HISTORIA DE USUARIO MODIFICAR LA EVALUACIÓN DEL TRABAJO DE TESIS.....	69
TABLA 3.2. 24 HISTORIA DE USUARIO ELIMINAR EVALUACIONES.....	70
TABLA 3.3. 1 ESTIMACIÓN DE ESFUERZO POR HISTORIA DE USUARIO.....	71
TABLA 3.3. 2 EQUIPO DE DESARROLLO1.....	73
TABLA 3.3. 3 PLAN DE ENTREGAS.....	74

TABLA 4.2 1 HISTORIAS DE USUARIO IMPLEMENTADAS EN LA PRIMERA ITERACIÓN.....	75
TABLA 4.2 2 TAREA #1 DE LA HISTORIA DE USUARIO AUTENTICAR USUARIO.....	75
TABLA 4.2 3 TAREA #2 DE LA HISTORIA DE USUARIO AUTENTICAR USUARIO.....	76
TABLA 4.2 4 TAREA #3 DE LA HISTORIA DE USUARIO AUTENTICAR USUARIO.....	76
TABLA 4.2 5 TAREA #1 DE LA HISTORIA DE USUARIO ELIMINAR USUARIO.....	77
TABLA 4.2 6 TAREA #2 DE LA HISTORIA DE USUARIO ELIMINAR USUARIO.....	77
TABLA 4.2 7 TAREA #3 DE LA HISTORIA DE USUARIO ELIMINAR USUARIO.....	77
TABLA 4.2 8 TAREA #1 DE LA HISTORIA DE USUARIO CREAR COMITÉ DE TESIS.....	78
TABLA 4.2 9 TAREA #2 DE LA HISTORIA DE USUARIO CREAR COMITÉ DE TESIS.....	78
TABLA 4.2 10 TAREA #3 DE LA HISTORIA DE USUARIO CREAR COMITÉ DE TESIS.....	79
TABLA 4.2 11 TAREA #1 DE LA HISTORIA DE USUARIO MODIFICAR COMITÉ DE TESIS.....	79
TABLA 4.2 12 TAREA #2 DE LA HISTORIA DE USUARIO MODIFICAR COMITÉ DE TESIS.....	79
TABLA 4.2 13 TAREA #3 DE LA HISTORIA DE USUARIO MODIFICAR COMITÉ DE TESIS.....	80
TABLA 4.2 14 TAREA #1 DE LA HISTORIA DE USUARIO ELIMINAR COMITÉ DE TESIS.....	80
TABLA 4.2 15 TAREA #2 DE LA HISTORIA DE USUARIO ELIMINAR COMITÉ DE TESIS.....	80
TABLA 4.2 16 TAREA #3 HISTORIA DE USUARIO ELIMINAR COMITÉ DE TESIS.....	81
TABLA 4.3. 1 HISTORIAS DE USUARIO IMPLEMENTADAS EN LA SEGUNDA ITERACIÓN.....	81
TABLA 4.3. 2 TAREA #1 HISTORIA DE USUARIO CREAR TRIBUNAL.....	82
TABLA 4.3. 3 TAREA #2 HISTORIA DE USUARIO CREAR TRIBUNAL.....	82
TABLA 4.3. 4 TAREA #3 HISTORIA DE USUARIO CREAR TRIBUNAL.....	82
TABLA 4.3. 5 TAREA #1 HISTORIA DE USUARIO MODIFICAR TRIBUNAL.....	83
TABLA 4.3. 6 TAREA #2 HISTORIA DE USUARIO MODIFICAR TRIBUNAL.....	83
TABLA 4.3. 7 TAREA #3 HISTORIA DE USUARIO MODIFICAR TRIBUNAL.....	84
TABLA 4.3. 8 TAREA #4 HISTORIA DE USUARIO MODIFICAR TRIBUNAL.....	84
TABLA 4.3. 9 TAREA #1 HISTORIA DE USUARIO ELIMINAR TRIBUNAL.....	84
TABLA 4.3. 10 TAREA #2 HISTORIA DE USUARIO ELIMINAR TRIBUNAL.....	85
TABLA 4.3. 11 TAREA #3 HISTORIA DE USUARIO ELIMINAR TRIBUNAL.....	85
TABLA 4.3. 12 TAREA #1 HISTORIA DE USUARIO CRONOGRAMA.....	85
TABLA 4.3. 13 TAREA #2 HISTORIA DE USUARIO CRONOGRAMA.....	86

TABLA 4.4. 1 HISTORIAS DE USUARIOS IMPLEMENTADAS EN LA TERCERA ITERACIÓN.	86
TABLA 4.4. 2 TAREA #1 HISTORIA DE USUARIO PUBLICAR TEMA DE TESIS.....	87
TABLA 4.4. 3 TAREA #2 HISTORIA DE USUARIO PUBLICAR TEMA DE TESIS.....	87
TABLA 4.4. 4 TAREA #1 HISTORIA DE USUARIO ELIMINAR TEMA DE TESIS.....	88
TABLA 4.4. 5 TAREA #2 HISTORIA DE USUARIO ELIMINAR TEMA DE TESIS.....	88
TABLA 4.4. 6 TAREA #3 HISTORIA DE USUARIO ELIMINAR TEMA DE TESIS.....	88
TABLA 4.4. 7 TAREA #1 HISTORIA DE USUARIO MODIFICAR TEMA DE TESIS.....	89
TABLA 4.4. 8 TAREA #2 HISTORIA DE USUARIO MODIFICAR TEMA DE TESIS.....	89
TABLA 4.4. 9 TAREA #3 HISTORIA DE USUARIO MODIFICAR TEMA DE TESIS.....	90
TABLA 4.4. 10 TAREA #1 HISTORIA DE USUARIO APROBAR TEMA DE TESIS.....	90
TABLA 4.4. 11 TAREA #2 HISTORIA DE USUARIO APROBAR TEMA DE TESIS.....	90
TABLA 4.4. 12 TAREA #3 HISTORIA DE USUARIO APROBAR TEMA DE TESIS.....	91
TABLA 4.4. 13 TAREA #4 HISTORIA DE USUARIO APROBAR TEMA DE TESIS.....	91
TABLA 4.4. 14 TAREA #1 HISTORIA DE USUARIO ASIGNAR TEMA DE TESIS.....	92
TABLA 4.4. 15 TAREA #2 HISTORIA DE USUARIO ASIGNAR TEMA DE TESIS.....	92
TABLA 4.4. 16 TAREA #3 HISTORIA DE USUARIO ASIGNAR TEMA DE TESIS.....	92
TABLA 4.4. 17 TAREA #1 HISTORIA DE USUARIO EVALUAR TRABAJO DE TESIS.....	93
TABLA 4.4. 18 TAREA #2 HISTORIA DE USUARIO EVALUAR TRABAJO DE TESIS.....	93
TABLA 4.4. 19 TAREA #3 HISTORIA DE USUARIO EVALUAR TRABAJO DE TESIS.....	93
TABLA 4.4. 20 TAREA #4 HISTORIA DE USUARIO EVALUAR TRABAJO DE TESIS.....	94
TABLA 4.4. 21 TAREA #1 HISTORIA DE USUARIO MODIFICAR LA EVALUACIÓN DE LOS TRABAJOS DE TESIS.....	94
TABLA 4.4. 22 TAREA #2 HISTORIA DE USUARIO MODIFICAR LA EVALUACIÓN DE LOS TRABAJOS DE TESIS.....	95
TABLA 4.4. 23 TAREA #2 HISTORIA DE USUARIO MODIFICAR LA EVALUACIÓN DE LOS TRABAJOS DE TESIS.....	95
TABLA 4.4. 24 TAREA #1 HISTORIA DE USUARIO ELIMINAR EVALUACIÓN.....	95
TABLA 4.4. 25 TAREA #2 HISTORIA DE USUARIO ELIMINAR EVALUACIÓN.....	96
TABLA 4.4. 26 TAREA #3 HISTORIA DE USUARIO ELIMINAR EVALUACIÓN.....	96
TABLA 4.5. 1 HISTORIAS DE USUARIOS IMPLEMENTADAS EN LA CUARTA ITERACIÓN.	97
TABLA 4.5. 2 TAREA #1 HISTORIA DE USUARIO LISTAR PROFESORES.	97
TABLA 4.5. 3 TAREA #2 HISTORIA DE USUARIO LISTAR PROFESORES.	97

TABLA 4.5. 4 TAREA #3 HISTORIA DE USUARIO LISTAR PROFESORES.	98
TABLA 4.5. 5 TAREA #1 HISTORIA DE USUARIO ADICIONAR PROFESORES.	98
TABLA 4.5. 6 TAREA #2 HISTORIA DE USUARIO ADICIONAR PROFESORES.	98
TABLA 4.5. 7 TAREA #1 HISTORIA DE USUARIO ELIMINAR PROFESORES.	99
TABLA 4.5. 8 TAREA #2 HISTORIA DE USUARIO ELIMINAR PROFESORES.	99
TABLA 4.5. 9 TAREA #1 HISTORIA DE USUARIO LISTAR GRUPOS DE QUINTO AÑO.....	100
TABLA 4.5. 10 TAREA #2 HISTORIA DE USUARIO LISTAR GRUPOS DE QUINTO AÑO.....	100
TABLA 4.5. 11 TAREA #3 HISTORIA DE USUARIO LISTAR GRUPOS DE QUINTO AÑO.....	100
TABLA 4.5. 12 TAREA #1 HISTORIA DE USUARIO ADICIONAR ESTUDIANTES.....	101
TABLA 4.5. 13 TAREA #2 HISTORIA DE USUARIO ADICIONAR ESTUDIANTES.....	101
TABLA 4.5. 14 TAREA #1 HISTORIA DE USUARIO ELIMINAR ESTUDIANTES.....	102
TABLA 4.5. 15 TAREA #2 HISTORIA DE USUARIO ELIMINAR ESTUDIANTES.....	102
TABLA 4.5. 16 TAREA #1 HISTORIA DE USUARIO LISTAR ESTUDIANTES SIN TESIS.....	102
TABLA 4.5. 17 TAREA #2 HISTORIA DE USUARIO LISTAR ESTUDIANTES SIN TESIS.....	103
TABLA 5.1. 1 ENTRADAS EXTERNAS.....	87
TABLA 5.1. 2 SALIDAS EXTERNAS.	88
TABLA 5.1. 3 FICHEROS INTERNOS.	88
TABLA 5.1. 4 PETICIONES.	88
TABLA 5.1. 5 INTERFACES EXTERNAS.....	89
TABLA 5.1. 6 PUNTOS DE FUNCIÓN DESAJUSTADOS.....	89
TABLA 5.2. 1 CARACTERÍSTICAS.....	89
TABLA 5.2. 2 FACTORES DE ESCALA.	90
TABLA 5.2. 3 MULTIPLICADORES DE ESFUERZO.....	90
TABLA 5.2. 4 RESULTADOS.....	91
ANEXO II TABLA 4.7. 1 PRUEBAS DE ACEPTACIÓN PARA LA HISTORIA DE USUARIO AUTENTICAR USUARIO.....	102
ANEXO II TABLA 4.7. 2 PRUEBAS DE ACEPTACIÓN PARA LA HISTORIA DE USUARIO AUTENTICAR USUARIO.....	102
ANEXO II TABLA 4.7. 3 PRUEBAS DE ACEPTACIÓN PARA LA HISTORIA DE USUARIO ELIMINAR USUARIO.....	103

ANEXO II TABLA 4.7. 4 PRUEBAS DE ACEPTACIÓN PARA LA HISTORIA DE USUARIO CREAR COMITÉ DE TESIS.....	103
ANEXO II TABLA 4.7. 5 PRUEBAS DE ACEPTACIÓN PARA LA HISTORIA DE USUARIO MODIFICAR COMITÉ DE TESIS.....	104
ANEXO II TABLA 4.7. 6 PRUEBAS DE ACEPTACIÓN PARA LA HISTORIA DE USUARIO ELIMINAR COMITÉ DE TESIS.....	104
ANEXO II TABLA 4.7. 7 PRUEBAS DE ACEPTACIÓN PARA LA HISTORIA DE USUARIO CREAR TRIBUNAL.....	105
ANEXO II TABLA 4.7. 8 PRUEBAS DE ACEPTACIÓN PARA LA HISTORIA DE USUARIO MODIFICAR TRIBUNAL.....	105
ANEXO II TABLA 4.7. 9 PRUEBAS DE ACEPTACIÓN PARA LA HISTORIA DE USUARIO ELIMINAR TRIBUNAL.....	106
ANEXO II TABLA 4.7. 10 PRUEBAS DE ACEPTACIÓN PARA LA HISTORIA DE USUARIO PUBLICAR TEMA DE TESIS.....	106
ANEXO II TABLA 4.7. 11 PRUEBAS DE ACEPTACIÓN PARA LA HISTORIA DE USUARIO ELIMINAR TEMA DE TESIS.....	107
ANEXO II TABLA 4.7. 12 PRUEBAS DE ACEPTACIÓN PARA LA HISTORIA DE USUARIO MODIFICAR TEMA DE TESIS.....	107
ANEXO II TABLA 4.7. 13 PRUEBAS DE ACEPTACIÓN PARA LA HISTORIA DE USUARIO APROBAR TEMA DE TESIS.....	108
ANEXO II TABLA 4.7. 14 PRUEBAS DE ACEPTACIÓN PARA LA HISTORIA DE USUARIO ASIGNAR TEMA DE TESIS.....	108
ANEXO II TABLA 4.7. 15 PRUEBAS DE ACEPTACIÓN PARA LA HISTORIA DE USUARIO EVALUAR TRABAJO DE TESIS.....	109
ANEXO II TABLA 4.7. 16 PRUEBAS DE ACEPTACIÓN PARA LA HISTORIA DE USUARIO LISTAR PROFESORES.....	109
ANEXO II TABLA 4.7. 17 PRUEBAS DE ACEPTACIÓN PARA LA HISTORIA DE USUARIO ADICIONAR PROFESOR.....	110
ANEXO II TABLA 4.7. 18 PRUEBAS DE ACEPTACIÓN PARA LA HISTORIA DE USUARIO ELIMINAR PROFESOR.....	110
ANEXO II TABLA 4.7. 19 PRUEBAS DE ACEPTACIÓN PARA LA HISTORIA DE USUARIO LISTAR GRUPOS DE QUINTO AÑO..	111
ANEXO II TABLA 4.7. 20 PRUEBAS DE ACEPTACIÓN PARA LA HISTORIA DE USUARIO ADICIONAR ESTUDIANTE	111
ANEXO II TABLA 4.7. 21 PRUEBAS DE ACEPTACIÓN PARA LA HISTORIA DE USUARIO ELIMINAR ESTUDIANTE	112
ANEXO II TABLA 4.7. 22 PRUEBAS DE ACEPTACIÓN PARA LA HISTORIA DE USUARIO LISTAR ESTUDIANTES SIN TESIS.....	112

ÍNDICE DE FIGURAS

FIGURA 2.4.5. 1 ARQUITECTURA DEL SISTEMA PROPUESTO.....	25
ANEXO I FIGURA 4.6. 1 DIAGRAMA DE CLASES DE ESTEREOTIPOS WEB.....	98
ANEXO I FIGURA 4.6. 2 DIAGRAMA DE CLASES PERSISTENTES.....	99
ANEXO I FIGURA 4.6. 3 DIAGRAMA DE LA BASE DE DATOS.	100
ANEXO I FIGURA 4.6. 4 DIAGRAMA DE DESPLIEGUE DEL SISTEMA.....	101

INTRODUCCION

En la facultad (8) se encuentra implementado un sistema que permite mantener actualizados tanto a estudiantes como profesores del desarrollo de tesis. Entre la información que maneja se encuentra, nombre de las tesis, autores que la desarrollan, tutores, tribunales de defensa, y diferentes reportes con mucha información que será útil tanto para profesores como para estudiantes. Sin embargo este sistema se encuentra implementado en Access por lo que la información no se encuentra publicada para ser utilizada por todo el que la necesite. Para mantener informado a los profesores y estudiantes es necesario imprimir volantes, los que, en caso de ocurrir algún cambio, dejarían de tener validez y se tendrán que imprimir cada vez que ocurra alguna modificación.

Tomando como base lo referido anteriormente, surge un **problema científico** ¿Cómo centralizar la información de la gestión y documentación del proceso de tesis en la facultad 8?

El **objeto de estudio** es el Proceso manual y automatización de gestión de tesis de la facultad 8.

El **campo de acción** esta dirigido al Proceso de automatización de la información que permita el manejo de los datos referentes a las tesis en la facultad 8.

Objetivo General:

Desarrollar una aplicación Web que permita el manejo de la información de las tesis en la facultad 8.

Objetivos Específicos:

- Estudiar la información que se encuentra en Access.
- Migrar la información existente en Access, a una aplicación Web que pueda ser accedida desde cualquier parte de la universidad.
- Realizar el análisis, diseño e implementación de la aplicación.

Idea a defender:

Si se implementa una aplicación Web para el manejo de los datos referentes a las tesis de la facultad 8, entonces podremos encontrar esta información disponible desde cualquier sitio de la UCI.

Para lograr el objetivo propuesto se han trazado una serie de **tareas de Investigación:**

1. Realizar un estudio del estado del arte teniendo en cuenta el trabajo ya realizado en Access.
2. Realizar un estudio sobre las metodologías posibles a utilizar.
3. Realizar un estudio de las principales tendencias y tecnologías actuales a utilizar.
4. Análisis, diseño e implementación del sistema.
5. Análisis de costos.

Capítulo I. Fundamentación teórica

1.0 Introducción.

En el presente capítulo se realizará un análisis de la práctica de integración continua como parte de la metodología de desarrollo *Extreme Programming* (XP), su aplicación a nivel mundial, en Cuba y en la Universidad de las Ciencias Informáticas (UCI). Se abordarán también las tecnologías, herramientas y metodología utilizadas para el desarrollo de la propuesta.

1.2 Estado del arte.

Proceso manual del desarrollo de tesis en la facultad 8.

El proceso de gestión de las tesis de forma manual es algo complicado, esto se debe a la gran cantidad de información que maneja el planificador y al elevado número de restricciones que debe tener en cuenta. En el momento de otorgar los temas de tesis la facultad le puede asignar una tesis, o el estudiante puede proponer un tema, el cual es analizado por un tribunal a nivel UCI y el mismo decide si el tema tiene alcance o no y si no existe una tesis que responda a los mismos objetivos que esta.

Este planificador conforma los tribunales de tesis que calificarán las mismas, poniéndose de acuerdo con los profesores, los tribunales se distribuyen por áreas de desarrollo (Calidad, Software Educativo, Software de Gestión, Aplicación Web, Gestión de Proyecto, Técnicas de Programación, Transferencias Tecnológicas (Servicios), Bases de Datos, Ingeniería de requerimientos, Inteligencia Artificial y Capacitación y Formación), cada uno atiende las tesis del área que le corresponde para lograr una mayor organización y tener un mejor control de las mismas. Además estas tesis pertenecen a diferentes entidades como son (UCI, Softel, FAR, FAC.8, MIC, etc.)

Para la asignación de los tutores por tesis, la planificadora hace un levantamiento de todos los profesores que cumplan con las condiciones para tutorar y va asignado uno o varios tutores por tema además de estos pueden asignar cotutores a las tesis que lo necesiten. Pero además si el estudiante propuso su tema de tesis y fue aceptado por el tribunal, este puede también proponer su tutor.

Algo complicado además, es establecer la planificación de las evaluaciones de las tesis, estas son, los cortes, la predefensa, y la exposición final del trabajo. Para cada evaluación se debe designar un local y una hora, la cual debe ser informada con antelación a los profesores para verificar que estos no coincidan con tutorías y tribunales a la misma hora y de ser así poder arreglar el calendario para que

no ocurran inconvenientes con el desarrollo de los trabajos. Las evaluaciones de cada corte deben ser guardadas al igual que los errores que se le encuentren a cada trabajo para verificar que estos sean arreglados.

Proceso automatizado del desarrollo de tesis en la facultad 8.

Actualmente la facultad 8 cuenta una aplicación desarrollada en Access que permite almacenar toda la información referente al proceso de tesis. En esta aparece un listado de los estudiantes de la facultad acompañado del tema de tesis de cada uno, así como el o los tutores que le corresponde a cada tesis. Además de esto la aplicación guarda el listado de profesores que conforma cada tribunal y el área de desarrollo de cada uno de estos. Por otro lado se lleva el control de las evaluaciones que ha tenido cada una de las tesis en las etapas propuestas y si quedan algunas pendientes para darle un tratamiento especial. Además de toda la información anterior esta aplicación guarda los horarios y los locales en los cuales se desarrollarán las tesis así como el área de desarrollo y la entidad a la que pertenecen. A pesar de la existencia de esta aplicación es muy difícil, tanto para estudiantes como para profesores informarse de lo que va aconteciendo pues ocurren cambios repentinos y esta información no tiene una vía rápida de llegar a los interesados. Los profesores eran informados de los días de tribunales y las tutorías mediante papeles impresos, los cuales caducaban con tan solo hacer un pequeño cambio, por lo que esto no es factible y el personal no dispone de la información actualizada del proceso de tesis de la facultad 8.

1.3 Propuesta de diseño.

1.3.1 CMS (*Content Management System*).

✓ Definición

La definición de 'Sistema para la Gestión de Contenidos', desde el punto de vista de la lógica de Marketing, resulta particularmente interesante. En efecto, sugiere que sea el propio sistema (y no el humano) el sujeto de la Gestión. Una definición más apropiada podría ser la de "sistema de soporte a la gestión de contenidos" ya que, en realidad, son las estrategias de comunicación las que realmente llevan a gestionar contenidos de forma efectiva; los sistemas informáticos pueden a lo sumo proporcionar las herramientas necesarias para la publicación en línea, o bien incluir servicios de

soporte a la toma de decisiones por lo que a la gestión de contenidos se refiere. Por el contrario, la definición acrónica de CMS (Content Management System) se aplica generalmente para referirse a sistemas de publicación.

Un sistema de administración de contenido a menudo funciona en el servidor del sitio Web. Muchos sistemas proporcionan diferentes niveles de acceso dependiendo del usuario, variando si es el administrador, editor, o creador de contenido. El acceso al CMS es generalmente por vía del navegador, y a veces se requiere el uso de FTP para subir contenido, generalmente fotografías o audio.

Los creadores de contenido crean sus documentos en el sistema. Los editores comentan, aceptan o rechazan los documentos. El editor en jefe es responsable por publicar el trabajo en el sitio. El CMS controla y ayuda a manejar cada paso de este proceso, incluyendo las labores técnicas de publicar los documentos a uno o más sitios. En muchos sitios con CMS una sola persona hace el papel de creador y editor, los blogs generalmente funcionan de esta manera. (E.Rivera 2007)(Graphicsperu 2007)(Xavier Cuerda García and Julià Minguillón Alfonso 2007)(2007)

✓ Distintos tipos de CMS

Existen diferentes CMS. Los podemos agrupar según el tipo de sitio que permiten gestionar. A continuación se muestran los más representativos:

- Genéricos: Ofrecen la plataforma necesaria para desarrollar e implementar aplicaciones que den solución a necesidades específicas. Pueden servir para construir soluciones de gestión de contenidos, para soluciones de comercio electrónico, blogs, portales,... Ejemplos: Zope, OpenCMS, Typo3, Apache lenya.
- Foros: sitio que permite la discusión en línea donde los usuarios pueden reunirse y discutir temas en los que están interesados. Ejemplos: phpBB, SMF, MyBB.
- Blogs: Publicación de noticias o artículos en orden cronológico con espacio para comentarios y discusión. Ejemplos: Wordpress, Typo.
- Wikis: Sitio Web dónde todos los usuarios pueden colaborar en los artículos, aportando información o reescribiéndola. También permite espacio para discusiones. Indicado para material que irá evolucionando con el tiempo. Ejemplos: Mediawiki, Tikiwiki.
- eCommerce: Son Sitios Web para comercio electrónico. Ejemplo: osCommerce

- Portal: Sitio Web con contenido y funcionalidad diversa que sirve como fuente de información o como soporte a una comunidad. Ejemplos: PHPNuke, Postnuke, Joomla, Drupal, Plone, IcyPhoenix.
- Galería: Permite administrar y generar automáticamente un portal o sitio Web que muestra contenido audiovisual, normalmente imágenes. Ejemplo: Gallery.
- e-Learning: Sirve para la enseñanza de conocimientos. Los usuarios son los profesores y estudiantes, tenemos aulas virtuales donde se pone a disposición el material del curso. La publicación de un contenido por un profesor es la puesta a disposición de los estudiantes, en un aula virtual, de ese contenido.

Ejemplo: Moodle.

- Publicaciones digitales: son plataformas especialmente diseñadas teniendo en cuenta las necesidades de las publicaciones digitales, tales como periódicos, revistas, etc.

Ejemplo: ePrints.

(Wikipedia® 2007)

En este caso el que se utilizará para el desarrollo de la Aplicación Web para el control del Proceso de Tesis de Grado en la Facultad 8 será uno de tipo Portal en específico Drupal.

CMS seleccionado (Drupal).

Luego de analizar los tipos de CMS existentes se decidió utilizar Drupal para el desarrollo de la aplicación.

¿Qué es Drupal?

Drupal, es el equivalente fonético en inglés a la palabra neerlandesa druppel que significa gota. Fue creado en 1999 por Dries Buytaert.

Es un sistema de gestión de contenidos (CMS) para sitios Web. Permite publicar artículos, imágenes, u otros archivos y servicios añadidos como foros, encuestas, votaciones, blogs y administración de usuarios y permisos.

Es un sistema dinámico: el contenido textual de las páginas y otras configuraciones son almacenados en una base de datos y se editan utilizando un entorno Web.

Utiliza como servidor Web Apache principalmente, como lenguaje de páginas dinámicas emplea PHP y MySQL como gestor de base de datos relacional. Es un CMS multiplataforma ya que puede utilizarse en Linux, BSD, Mac OS X, Windows, Solaris entre otros.

Drupal cuenta además con un sinnúmero de ventajas que llevaron a su selección, entre ellas se encuentran:

- Es software libre.
- Contiene las principales características de un CMS en el núcleo del sistema.
- Gran comunidad de usuarios.
- Muy flexible (se adapta a casi cualquier cosa).
- Ampliable: cientos de módulos.
- Buena arquitectura de programación.
 - Seguro.
 - Buen rendimiento.

Características principales

Sistema de taxonomías (categorías) para archivar el contenido.

- Tipos de contenido personalizables.
- Se lleva bien con Google.
- Gestión completa de usuarios (roles).
- Personalización de la presentación del contenido en pantalla (bloques).
- Sistema de taxonomías (categorías) para archivar el contenido.

(Sitio Dupal 2007)

1.3.2 Lenguajes de programación Web.

La programación Web, parte de las siglas WWW, que significan World Wide Web o telaraña mundial. Para realizar una página con la programación Web, se deben tener claros tres conceptos fundamentales, los cuales son, el URL (Uniform Resource Locators), es un sistema con el cual se

localiza un recurso dentro de la red, este recurso puede ser una página Web, un servicio o cualquier otra cosa. En resumen el URL no es más que un nombre, que identifica una computadora, dentro de esa computadora un archivo que indica el camino al recurso que se solicita. A continuación se presentan varios lenguajes de programación que se utilizan para la construcción de sitios Web.

✓ JAVA

El lenguaje para la programación en Java, es un lenguaje orientado a objeto, de una plataforma independiente. Fue desarrollado por la compañía Sun Microsystems, con la idea original de usarlo para la creación de páginas WEB. Esta programación tiene muchas similitudes con el lenguaje C y C++.

El código generado por el compilador Java es independiente de la arquitectura: podría ejecutarse en un entorno UNIX, Mac o Windows. Esto es posible ya que el código generado por el compilador se ejecuta mediante una máquina virtual y por el procesador del ordenador directamente. Esto permite que los Applets de una Web pueda ejecutarlos cualquier máquina que se conecte a ella independientemente de qué sistema operativo emplee, siempre y cuando el ordenador en cuestión tenga instalada una máquina virtual de Java.

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 pagina HTML en un servidor Web. Por lo general los Applets son programas pequeños y de propósitos específicos.(Computación Aplicada al Desarrollo SA de CV 2007)

PERL

Estructuralmente, Perl está basado en un estilo de bloques como los del C o AWK, y fue ampliamente adoptado por su destreza en el procesado de texto y no tener ninguna de las limitaciones de los otros lenguajes de script.

El manual Unix dice:

Perl es un lenguaje de propósito general, originalmente desarrollado para la manipulación de texto y que ahora es utilizado para un amplio rango de tareas incluyendo administración de sistemas, desarrollo Web, programación en red, desarrollo de GUI y más.

Se previó que fuera práctico (facilidad de uso, eficiente, completo) en lugar de hermoso (pequeño, elegante, mínimo).

✓ Sus principales características son:

- La estructura completa de Perl deriva ampliamente del lenguaje C.
- Es fácil de usar.
- Soporta tanto la programación estructurada como la programación orientada a objetos y la programación funcional.
- Tiene incorporado un poderoso sistema de procesamiento de texto y una enorme colección de módulos disponibles.
- Perl es un lenguaje imperativo, con variables, expresiones, asignaciones, bloques de código delimitados por llaves, estructuras de control y subrutinas.
- En Perl 5, se añadieron características para soportar estructuras de datos complejas, funciones de primer orden (p. e. clausuras como valores) y un modelo de programación orientada a objetos. Éstos incluyen referencias, paquetes y una ejecución de métodos basada en clases y la introducción de variables de ámbito léxico, que hizo más fácil escribir código robusto (junto con el pragma `strict`).
- Una característica principal introducida en Perl 5 fue la habilidad de empaquetar código reutilizable como módulos.

Larry Wall indicó más adelante que "la intención del sistema de módulos de Perl 5 era apoyar el crecimiento de la cultura Perl en vez del núcleo de Perl".

- Todas las versiones de Perl hacen el tipificado automático de datos y la gestión de memoria.
- El intérprete conoce el tipo y requerimientos de almacenamiento de cada objeto en el programa; reserva y libera espacio para ellos según sea necesario.

- Las conversiones legales de tipo se hacen de forma automática en tiempo de ejecución; las conversiones ilegales son consideradas errores fatales. (Kirrily "Skud" Robert <skud@cpan.org> 2007)

✓ PHP

PHP es un lenguaje de programación interpretado usado normalmente para la creación de páginas web dinámicas. PHP es un acrónimo recursivo que significa "**PHP** Hypertext **Pre**-processor" (inicialmente PHP Tools, o, *Personal Home Page Tools*). Actualmente también se puede utilizar para la creación de otros tipos de programas incluyendo aplicaciones con interfaz gráfica usando las bibliotecas Qt o GTK+.

La versión más reciente de PHP es la 5.2.5 (8 de noviembre de 2007), que incluye varias ventajas:

- Mejor soporte para la Programación Orientada a Objetos, que en versiones anteriores era extremadamente rudimentario, con PHP Data Objects.
- Mejoras de rendimiento.
- Mejor soporte para MySQL con extensión completamente reescrita.
- Mejor soporte a XML (XPath, DOM, etc.).
- Soporte nativo para SQLite.
- Soporte integrado para SOAP.
- Iteradores de datos.
- Manejo de excepciones.

(PHP en la Web 2007)

Lenguaje seleccionado (PHP)

Luego del análisis de los lenguajes anteriores se decide utilizar para el desarrollo del sistema el lenguaje PHP ya que tiene total compatibilidad con el CMS que será usado para el desarrollo de nuestra aplicación, incluso el CMS trae dentro de algunos de sus módulos de instalación el PHP y específicamente se usará el PHP 5.

Además de esto PHP tiene otras ventajas, entre las que se encuentran:

- Es un lenguaje multiplataforma.
- Capacidad de conexión con la mayoría de los manejadores de base de datos que se utilizan en la actualidad, destaca su conectividad con MySQL.
- Capacidad de expandir su potencial utilizando la enorme cantidad de módulos (llamados ext's o extensiones).
- Posee una amplia documentación en su página oficial, entre la cual se destaca que todas las funciones del sistema están explicadas y ejemplificadas en un único archivo de ayuda. Además de ser ampliamente usado en la universidad e impartido en el programa docente de la facultad.
- Es libre, por lo que se presenta como una alternativa de fácil acceso para todos.
- Permite las técnicas de Programación Orientada a Objetos.
- Biblioteca nativa de funciones sumamente amplia e incluida.
- No requiere definición de tipos de variables.
- Tiene manejo de excepciones.
- Además PHP está basado en el lenguaje C++ y la sintaxis usada es muy similar a C/C++ el cual es considerado aún el mejor lenguaje de programación por muchos programadores.

1.3.3 Servidores Web.

✓ Apache

El servidor Apache se desarrolla dentro del proyecto HTTP Server (httpd) de la Apache Software Foundation.

Apache presenta entre otras características mensajes de error altamente configurables, bases de datos de autenticación y negociado de contenido, pero fue criticado por la falta de una interfaz gráfica que ayude en su configuración.

Apache tiene amplia aceptación en la red: en el 2005, Apache fue el servidor HTTP más usado, siendo el servidor empleado en el 48% de los sitios web en el mundo. Sin embargo ha sufrido un descenso en su cuota de mercado en los últimos años.

La mayoría de las vulnerabilidades de la seguridad descubiertas y resueltas tan sólo pueden ser aprovechadas por usuarios locales y no remotamente. Sin embargo, algunas se pueden accionar remotamente en ciertas situaciones, o explotar por los usuarios locales malévolos en las disposiciones

de recibimiento compartidas que utilizan PHP como módulo de Apache.(The Apache Software Foundation. 2007)

✓ Lighttpd

Lighttpd es un servidor web diseñado para ser rápido, seguro, flexible, y fiel a los estándares. Está optimizado para entornos donde la velocidad es muy importante, y por eso consume menos CPU y memoria RAM que otros servidores. Por todo lo que ofrece, *lighttpd* es apropiado para cualquier servidor que tenga problemas de carga.

Lighttpd es software libre y se distribuye bajo la licencia BSD. Funciona en GNU/Linux y UNIX de forma oficial. Para Microsoft Windows actualmente hay una distribución conocida como Lighttpd For Windows mantenida por *Kevin Worthington*.

Características

- Virtual hosting (alojar varios dominios en la misma IP).
- CGI, SCGI y FastCGI.
- Soporte para PHP, Ruby, y otros.
- Entorno chroot.
- Cifrado SSL.
- Compresión (gzip, bzip2.).
- Autenticación (LDAP, htpasswd, otros).
- Server Side Includes.
- Consumo de memoria constante.
- Redirecciones HTTP, y reescrituras de URL.
- Puede enviar partes de un fichero (rangos).
- Puede usar select() o poll().
- También permite otros sistemas de *notificación de eventos* como kqueue y epoll.
- Hace estadísticas mediante RRDtool.
- Muestra un listado de ficheros cuando se entra a un directorio sin *index.html*.
- Redirección condicional.
- Permite módulos externos.
- Cache Meta Language.
- Acepta parte de WebDAV.

(lighttpd fly light 2007)

✓ IIS

Internet Information Services, IIS, es una serie de servicios para los ordenadores que funcionan con Windows. Los servicios que ofrece son: FTP, SMTP, NNTP y HTTP/HTTPS.

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

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

Versiones

- IIS 1.0, Windows NT 3.51 Service Pack 3.
- IIS 2.0, Windows NT 4.0.
- IIS 3.0, Windows NT 4.0 Service Pack 3.
- IIS 4.0, Windows NT 4.0 Option Pack.
- IIS 5.0, Windows 2000.
- IIS 5.1, Windows XP Professional.
- IIS 6.0, Windows Server 2003 y Windows XP Professional x64 Edition.
- IIS 7.0, Windows Vista y Windows Server 2008.

(HTML.it s.r.l. - Design 2007)

✓ Thttpd

Características

- *Simple*, por que esto maneja sólo el mínimo necesario para poner en práctica el protocolo HTTP, algunas veces un poco más que el mínimo.
- *Pequeño*, porque esto también tiene un pequeño tamaño de período de explotación, ya que esto no se divide en dos partes y es muy cuidadoso sobre la asignación de memoria.

- *Portátil*, porque esto se compila limpiamente sobre la mayoría de sistemas operativos, expresamente incluyendo FreeBSD, SunOS 4, Solaris 2, BSD/OS, Linux, OSF.
- *Rápido*, porque en el empleo típico es sobre todo más rápido que los mejores servidores “destacados” (APACHE), y bajo la carga extrema es mucho más rápido.
- *Seguro*, porque este se extiende a grandes longitudes para proteger el servidor WEB contra ataques otros sitios.

Ventajas

El administrador puede decidir restringir la transferencia de archivos de imagen JPEG en la mayor parte a 20 kilobytes por segundo. Esto impide a la conexión hacerse saturado de modo que el servidor todavía sea sensible bajo la carga pesada, con la compensación que reducen (obligan) la velocidad de transferencia de archivo.

Los promedios de carga se caen debido a la reducción de la transferencia gráfica gracias a THTTPD.

Desventajas

No posee las mismas aplicaciones que se pueden obtener de un software estándar como lo es el Apache. (THTTPD 2007)

Servidor Web seleccionado (Apache)

Luego de hacerse un profundo análisis de Apache, Lighttpd, IIS y Thttpd se ha decidido que para el entorno a desarrollar el mejor servidor Web es el Apache por las siguientes razones:

La mayoría de las vulnerabilidades de la seguridad descubiertas y resueltas tan sólo pueden ser aprovechadas por usuarios locales y no remotamente. Sin embargo, algunas se pueden accionar remotamente en ciertas situaciones, o explotar por los usuarios locales malévolos en las disposiciones de recibimiento compartidas que utilizan PHP como módulo de Apache.

Ventajas

- El servidor HTTP Apache es un software libre.
- Es muy usado en la facultad 8 y en toda la universidad.
- Amplia disposición de Documentos e información en general para el trabajo con este software en la universidad.

- Modular.
- Open source.
- Multi-plataforma.
- Extensible.
- Popular (fácil conseguir ayuda/soporte en Internet y otros sitios).
- Presenta entre otras características mensajes de error altamente configurables.
- Apache tiene amplia aceptación en la red.
- El CMS que será usado en el desarrollo de la aplicación en cuestión, propone como uno de los servidores Web al Apache, y entre lo propuestos este es el mejor candidato.

1.3.4 Gestor de base de datos.

✓ MySQL

MySQL es un sistema de gestión de base de datos relacional, multihilo y multiusuario con más de seis millones de instalaciones.

Existen varias APIs que permiten, a aplicaciones escritas en diversos lenguajes de programación, acceder a las bases de datos MySQL, incluyendo C, C++, C#, Pascal, Delphi (via dbExpress), Eiffel, Smalltalk, Java (con una implementación nativa del driver de Java), Lisp, Perl, PHP, Python, Ruby, Gambas, REALbasic (Mac), FreeBASIC, y Tcl; cada uno de estos utiliza una API específica. También existe un interfaz ODBC (Open Data base Connectivity), llamado MyODBC que permite a cualquier lenguaje de programación que soporte ODBC comunicarse con las bases de datos MySQL.

MySQL es muy utilizado en aplicaciones web como MediaWiki o Drupal, en plataformas (Linux/Windows-Apache-MySQL-PHP/Perl/Python), y por herramientas de seguimiento de errores como Bugzilla. Su popularidad como aplicación Web está muy ligada a PHP, que a menudo aparece en combinación con MySQL. En aplicaciones Web hay baja concurrencia en la modificación de datos y en cambio el entorno es intensivo en lectura de datos, lo que lo hace ideal para este tipo de aplicaciones.

Características distintivas

- Múltiples motores de almacenamiento (MyISAM, Merge, InnoDB, BDB, Memory/heap, MySQL Cluster, Federated, Archive, CSV, Blackhole y Example en 5.x), permitiendo al usuario escoger la que sea más adecuada para cada tabla de la base de datos.
 - Agrupación de transacciones, reuniendo múltiples transacciones de varias conexiones para incrementar el número de transacciones por segundo. (MySQL 2007)(MySQL AB. 2007)
- ✓ PostgreSQL

Es un servidor de base de datos relacional orientada a objetos de software libre, liberado bajo la licencia BSD.

Este presenta una alta concurrencia ya que mediante un sistema denominado MVCC (Acceso concurrente multiversión, por sus siglas en inglés) PostgreSQL permite que mientras un proceso escribe en una tabla, otros accedan a la misma tabla sin necesidad de bloqueos. Cada usuario obtiene una visión consistente de lo último a lo que se le hizo *commit*. Esta estrategia es superior al uso de bloqueos por tabla o por filas común en otras bases, eliminando la necesidad del uso de bloqueos explícitos.(PostgreSQL 2007)

✓ Firebird

Es un sistema de administración de base de datos relacional (o RDBMS) SQL de código abierto, basado en la versión 6 de Interbase, cuyo código fue liberado por Borland en 2000. Su código fue reescrito de C a C++.

Es multiplataforma, y actualmente puede ejecutarse en los sistemas operativos: Linux, HP-UX, FreeBSD, Mac OS, Solaris y Microsoft Windows. Presenta un ejecutable pequeño, con requerimientos de hardware bajos. Utiliza la arquitectura Cliente/Servidor sobre protocolo TCP/IP y otros (embedded) además de que soporta transacciones y claves foráneas.

Soporta plenamente el estándar SQL-92, tanto de sintaxis como de tipos de datos. Además tiene capacidad de almacenar elementos BLOB (*binary large objects*). (IBPhoenix 2007)

Servidor de Base de datos seleccionado (MySQL)

Luego del estudio de varios servidores de bases de datos se ha seleccionado para la realización del sistema MySQL debido a que:

- Es un sistema gestor de bases de datos “Open Source”.
- Presenta múltiples motores de almacenamiento, permitiendo al usuario escoger el que sea más adecuado para cada tabla de la base de datos.
- Agrupación de transacciones, reuniendo múltiples transacciones de varias conexiones para incrementar el número de transacciones por segundo.
- MySQL está escrito en una mezcla de C y C++.

Además de las facilidades antes mencionadas se decidió escoger este gestor pues es muy usado en la universidad por ser muy rápido en aplicaciones de este tipo y su compatibilidad con el CMS a usar.

1.3.5 Metodologías.

- ✓ Ingeniería de software

La Ingeniería de Software es una tecnología que indica “COMO” construir técnicamente un software: económico, fiable y que funcione eficientemente. Dentro de esta se encuentra la metodología que se va a utilizar para el desarrollo del proyecto.

- ✓ Razones de la Ingeniería de software

Estudiar ¿cuáles son las actividades que organizadas, haciendo uso racional de los recursos y apoyándose en técnicas y herramientas, logran la mayor eficiencia al construir un software?.

- ✓ Metodología

La rama de la **metodología**, dentro de la ingeniería de software, se encarga de elaborar estrategias de desarrollo de software que promuevan prácticas adaptativas en vez de predictivas; centradas en las personas o los equipos, orientadas hacia la funcionalidad y la entrega, de comunicación intensiva y que requieren implicación directa del cliente. (Metodologías De Desarrollo De Software 2007)

RUP

El **Proceso Unificado Racional** (*Rational Unified Process* en inglés, habitualmente resumido como **RUP**) es un proceso de desarrollo de software y junto con el Lenguaje Unificado de Modelado UML,

constituye la metodología estándar más utilizada para el análisis, implementación y documentación de sistemas orientados a objetos.

- El RUP está basado en 5 principios clave que son:

Adaptar el proceso.

Balancear prioridades.

Demostrar valor iterativamente.

Elevar el nivel de abstracción.

Enfocarse en la calidad.

- El ciclo de vida RUP

El RUP divide el proceso de desarrollo en ciclos, teniendo un producto final al culminar cada una de ellos, estos a la vez se dividen en fases que finalizan con un hito donde se debe tomar una decisión importante:

- **Inicio:** se hace un plan de fases, se identifican los principales casos de uso y se identifican los riesgos.
- **Elaboración:** se hace un plan de proyecto, se completan los casos de uso y se eliminan los riesgos.
- **Construcción:** se concentra en la elaboración de un producto totalmente operativo y eficiente y el manual de usuario.
- **Transición:** se instala el producto en el cliente y se entrena a los usuarios. Como consecuencia de esto suelen surgir nuevos requisitos a ser analizados.
- Los elementos de RUP son:
 - Actividades: Procesos que se determinan en cada iteración.
 - Trabajadores: Personas o entes involucradas en cada proceso.
 - Artefactos: Puede ser un documento, un modelo, o un elemento de modelo.

XP

La Programación Extrema es una metodología ligera de desarrollo de software que se basa en la simplicidad, la comunicación y la realimentación o reutilización del código desarrollado.

Las cuatro variables de esta metodología son:

1. Coste: Máquinas, especialistas y oficinas.
2. Tiempo: Total y de Entregas.
3. Calidad: Externa e Interna.
4. Alcance: Intervención del cliente.

XP surgió como respuesta y posible solución a los problemas derivados del cambio en los requerimientos. Se plantea como una metodología a emplear en proyectos de riesgo. Con ella se aumenta la productividad.(Escribano 2002)

Metodología seleccionada (XP)

Luego del estudio de las metodologías anteriores se decide utilizar XP principalmente por que el proyecto a desarrollar no será implementado orientado a objeto, es una de las metodologías de desarrollo de software más exitosas en la actualidad utilizadas para proyectos de corto plazo, corto equipo y cuyo plazo de entrega era ayer. La metodología consiste en una programación rápida o extrema, cuya particularidad es tener como parte del equipo, al usuario final, pues es uno de los requisitos para llegar al éxito del proyecto.

Características de XP, la metodología se basa en:

- **Pruebas Unitarias:** se basa en las pruebas realizadas a los principales procesos, de tal manera que adelantándonos en algo hacia el futuro, podamos hacer pruebas de las fallas que pudieran ocurrir. Es como si nos adelantáramos a obtener los posibles errores.
- **Refabricación:** se basa en la reutilización de código, para lo cual se crean patrones o modelos estándares, siendo más flexible al cambio.
- **Programación en pares:** una particularidad de esta metodología es que propone la programación en pares, la cual consiste en que dos desarrolladores participen en un

proyecto en una misma estación de trabajo. Cada miembro lleva a cabo la acción que el otro no está haciendo en ese momento. Es como el chofer y el copiloto: mientras uno conduce, el otro consulta el mapa.

¿Qué es lo que propone XP?

- Empieza en pequeño y añade funcionalidad con retroalimentación continua.
- El manejo del cambio se convierte en parte sustantiva del proceso.
- El costo del cambio no depende de la fase o etapa.
- No introduce funcionalidades antes que sean necesarias.
- El cliente o el usuario se convierte en miembro del equipo.

Derechos del Cliente.

- Decidir que se implementa.
- Saber el estado real y el progreso del proyecto.
- Añadir, cambiar o quitar requerimientos en cualquier momento.
- Obtener lo máximo de cada semana de trabajo.
- Obtener un sistema funcionando cada 3 o 4 meses.

Derechos del Desarrollador.

- Decidir como se implementan los procesos.
- Crear el sistema con la mejor calidad posible.
- Pedir al cliente en cualquier momento aclaraciones de los requerimientos.
- Estimar el esfuerzo para implementar el sistema.
- Cambiar los requerimientos sobre la base a nuevos descubrimientos.

Lo fundamental en este tipo de metodología es:

- La comunicación, entre los usuarios y los desarrolladores.
- La simplicidad, al desarrollar y codificar los módulos del sistema.
- La retroalimentación, concreta y frecuente del equipo de desarrollo, el cliente y los usuarios finales.

1.3.6 Plataforma.

Aplicación Web.

Una aplicación Web es un sistema informático que los usuarios utilizan accediendo a un servidor Web a través de Internet o de una intranet. Las aplicaciones Web son populares debido a la practicidad del navegador Web como cliente ligero. La facilidad para actualizar y mantener aplicaciones Web sin distribuir e instalar software en miles de potenciales clientes es otra razón de su popularidad. Aplicaciones como los webmails, wikis o tiendas en línea son ejemplos bien conocidos de aplicaciones Web.

Una ventaja principal es que deberían funcionar igual independientemente de la versión del sistema operativo instalado en el cliente. En vez de crear clientes para Windows, Mac OS X, GNU/Linux, y otros sistemas operativos, la aplicación es escrita una vez y es mostrada casi en todos lados.

➤ Ventajas:(2005)

- Desarrollo barato, sencillo y rápido.
- Acceso ubicuo, sin necesidad de distribución e, idealmente, con pocos requerimientos técnicos.
- Datos centralizados y fácil integración de datos de múltiples fuentes.
- Permiten el desarrollo de comunidades que dan valor a las aplicaciones (software social).
- Una empresa puede migrar de sistema operativo o cambiar el Hardware libremente sin afectar el funcionamiento de las aplicaciones de servidor.
- No se requieren complicadas combinaciones de Hardware/Software para utilizar estas aplicaciones. Solo un computador con un buen navegador Web.
- Actualizar o hacer cambios en el Software es sencillo y sin riesgos de incompatibilidades. Existe solo una versión en el servidor lo que implica que no hay que distribuirla entre los demás computadores. El proceso es rápido y limpio.
- Se facilita el trabajo a distancia. Se puede trabajar desde cualquier PC o computador portátil con conexión a Internet o a una red interna o privada.
- Al funcionar en un navegador, se requiere un conocimiento básico de informática para utilizar una aplicación Web.

Todas estas ventajas dejan claro el potencial de las aplicaciones Web. La utilización de ésta tecnología conlleva a reducir costos y complicaciones, y proporciona mayor libertad a la hora de realizar cualquier tipo de cambios.

Las aplicaciones Web utilizan la arquitectura cliente-servidor que es una forma específica de diseño de aplicaciones, aunque también se conoce con este nombre a los ordenadores en los que estas aplicaciones son ejecutadas. Por un lado, el cliente es el ordenador que se encarga de efectuar una petición o solicitar un servicio. El cliente no posee control sobre los recursos, sino que es el servidor el encargado de manejarlos. Por otro lado, el ordenador remoto que actúa como servidor evalúa la petición del cliente y decide aceptarla o rechazarla consecuentemente. Una vez que el servidor acepta el pedido la información requerida es suministrada al cliente que efectuó la petición, siendo este último el responsable de proporcionar los datos al usuario con el formato adecuado. El cliente y el servidor no tienen que estar necesariamente en ordenadores separados, sino que pueden ser programas diferentes que se ejecuten en el mismo ordenador. (Aplicaciones Web 2007)

Conclusiones

En este capítulo se ha hecho un análisis de todos los lenguajes, metodologías y herramientas propuestos para darle solución al sistema siendo elegidos los mejores candidatos para darle solución a la propuesta, después de un profundo análisis se ha llegado a la conclusión que es importante realizar una aplicación Web teniendo en cuenta la facilidad de acceso a la misma desde distintas partes.

Es de vital importancia el dominio de estas herramientas seleccionadas y analizar la mejor forma de aplicarlas, para desarrollar un sistema de máxima calidad que cumpla con los requisitos propuestos y de al cliente una versión del producto que satisfaga sus intereses.

Capítulo II. Características del Sistema

2.1 Introducción.

En este capítulo se realizará un análisis de las características del sistema a desarrollar, haciendo hincapié en la situación problemática que da origen al mismo.

2.2. Objeto de estudio.

Proceso manual y automatización de gestión de tesis de la facultad 8.

2.2.1 Situación Problemática.

En la facultad 8 se encuentra implementado un sistema que permite mantener actualizados tanto a estudiantes como profesores del desarrollo de tesis. Entre la información que maneja se encuentra, nombre de las tesis, autores que la desarrollan, tutores, tribunales de defensa, y diferentes reportes con mucha información que será útil tanto para profesores como para estudiantes. Sin embargo este sistema se encuentra implementado en Access por lo que la información no se encuentra publicada para ser utilizada por todo el que la necesite. Para mantener informado a los profesores y estudiantes es necesario imprimir volantes, los que, en caso de ocurrir algún cambio, dejarían de tener validez y se tendrán que imprimir cada vez que ocurra alguna modificación. Por otro lado, el comité de tesis tiene que reunirse con mucha frecuencia para manejar informaciones que podrían estar publicadas.

2.2.2 Problema

El problema fundamental es que no existe un sistema o aplicación dinámica que permita el control y centralización de los datos y documentos concernientes al proceso de tesis de grado en la Facultad 8. Por esta razón se desea construir una aplicación Web para automatizar estas tareas.

2.2.3 Objeto de Automatización

Durante el ciclo de desarrollo desde que se conforman las tesis de grado hasta su exposición y aprobación final existen varios procesos que deben ser automatizados, puesto que su ejecución de forma manual resulta más engorrosa y propensa a errores, además de consumir tiempo al personal encargado de realizar las tareas y recursos a la facultad, como son hojas y demás. Se desea desarrollar automatizar el control así como la centralización de los documentos concernientes al proceso de tesis de grado en la Facultad 8.

2.2.3 Información que se maneja

La información que se maneja son los documentos generados por el proceso de tesis de grado y las autoridades concernientes al proceso en cuestión.

2.2.4 Propuesta del Sistema

Se ha decidido darle solución al problema mediante una aplicación Web que facilite las tareas antes mencionadas en el menor tiempo posible y con la menor cantidad de contratiempos. Esta aplicación se realizará utilizando el CMS Drupal al cual se le implementará una aplicación que automatizará el proceso de tesis de grado, cumpliendo los requisitos pedidos por el usuario.

2.4.5 Arquitectura del sistema propuesto:

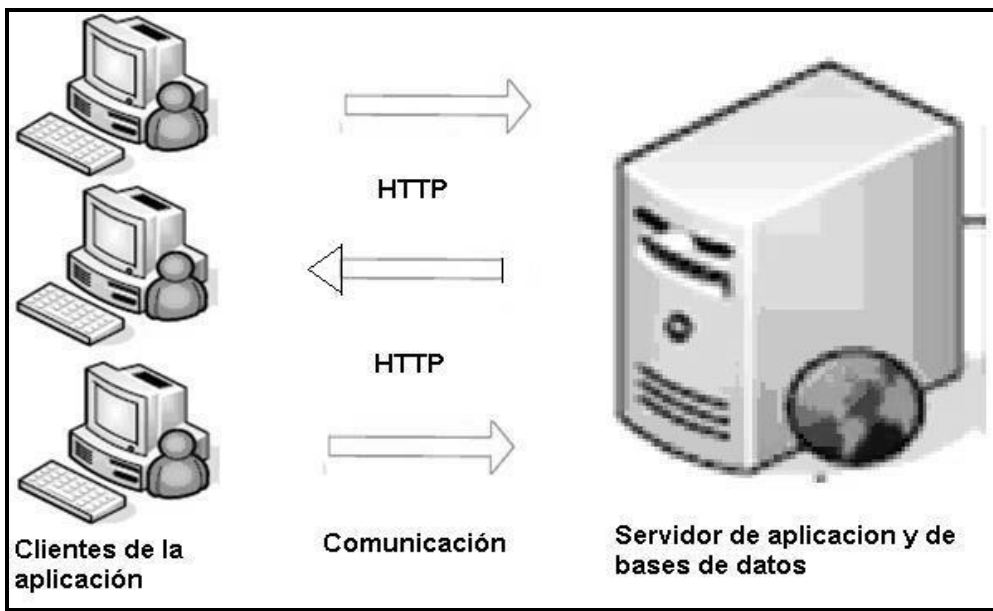


Figura 2.4.5. 1 Arquitectura del sistema propuesto

Aplicación Web para el control del Proceso de Tesis de Grado en la Facultad 8: Esta aplicación es la encargada de automatizar el trabajo de las entidades relacionadas con el proceso de desarrollo de tesis de grado así como la centralización de los documentos generados entorno del proceso en cuestión, esta aplicación permitirá publicar los temas de tesis propuestos por lo profesores, asignar dichos temas a los estudiantes de quinto año que aun no tienen un tema asignado, tener un seguimiento de la evaluación y evolución de las tesis así como de sus tesis, brindara informes relacionados con todo el proceso de tesis que pueden ser entre otros horarios de exposición, locales, tesis evaluadas, tesis sin evaluar, tribunales, tutores, etc.

2.4.6 Interfaz de usuario

- **Autenticar Usuario:** Interfaz mediante la cual el usuario se autentica en el sistema.
- **Usuarios:** Interfaz mediante la cual el administrador del sistema elimina o agrega un usuario a un rol determinado, retirándole o dándole los privilegios de este en la aplicación, en caso de ser un usuario sin privilegios de administración (Profesor o estudiante) esta interfaz no se mostrará.
- **Administrar:** Esta interfaz será manejada por el administrador del sistema, desde la misma se podrá agregar un usuario, darle o quitarle privilegios a un usuario, puede también eliminar o modificar un usuario determinado, en caso de ser un usuario sin privilegios de administración (Profesor o estudiante) esta interfaz no se mostrará.
- **Proponer tesis:** En esta interfaz el profesor podrá publicar un tema de tesis y modificar el mismo en caso de creerlo necesario mientras esta no sea aprobada por el comité de tesis, luego de esta haber sido aprobada el profesor ya no podrá efectuar ninguna función en la misma, es una interfaz sencilla y fácil de usar, esto es con el objetivo de que no solo puedan publicar un tema de tesis quien tenga conocimientos de informática sino cualquier otro profesor.
- **Revisar propuestas:** En esta interfaz se muestra el nombre del autor un resumen de la propuesta esta interfaz es manipulada por el jefe de comité de tesis quien revisa la propuesta en caso de ser aprobada se le asigna un área de desarrollo y se acepta, en el caso contrario se pone un comentario con un resumen de por qué fue rechazada, en ese caso el tema va a tesis desechadas, en caso de ser un usuario sin privilegios de administración (Profesor o estudiante) esta interfaz no se mostrara.
- **Tesis rechazadas:** En esta interfaz el profesor puede ver los errores de su propuesta corregirlos y volver a publicarlo para su próxima revisión.
- **Tesis libres:** Muestra un listado de las tesis aprobadas que aún no han sido asignadas.
- **Asignar tesis:** Esta interfaz será manipulada por el jefe de comité de tesis que es el encargado de, entre otras cosas, asignar los temas de tesis en la aplicación, en la misma se elige un tema de tesis de los del banco de tesis donde estarán los temas de tesis ya aprobados que aún no han sido asignados y se le asigna a los estudiantes que serán los encargados de su desarrollo, en caso de ser un usuario sin privilegios de administración (Profesor o estudiante) esta interfaz no se mostrará.
- **Tribunales:** Esta interfaz será manipulada por el jefe de comité de tesis, en la misma se podrá crear un tribunal, modificar un tribunal, así como eliminar el mismo, en caso de ser un usuario

sin privilegios de administración (Profesor o estudiante) esta interfaz se verá, solo mostrará información.

- **Comité de tesis:** Esta interfaz será manipulada por el jefe de comité de tesis, en la misma se podrá crear un comité de tesis, modificar un comité de tesis, y eliminar un comité de tesis, en caso de ser un usuario sin privilegios de administración (Profesor o estudiante) esta interfaz se mostrará pero solo con información.
- **Tesis evaluadas:** En esta interfaz se podrá llevar el seguimiento de las evaluaciones realizadas a un tema de tesis así como actualizar las mismas luego de un corte de tesis, en caso de ser un usuario sin privilegios de administración (Profesor o estudiante) esta interfaz se mostrará pero solo con información.
- **Editar evaluación:** En esta interfaz se podrán modificar las evaluaciones realizadas a un trabajo de tesis determinado, en caso de ser un usuario sin privilegios de administración (Profesor o estudiante) esta interfaz no se mostrará.
- **Eliminar evaluación:** En esta interfaz se podrán eliminar las evaluaciones realizadas a un trabajo de tesis determinado, en caso de ser un usuario sin privilegios de administración (Profesor o estudiante) esta interfaz no se mostrará.
- **Cronograma:** En esta interfaz el planificador creará el cronograma de trabajo para los distintos tribunales en cada corte de tesis, en caso de ser un usuario sin privilegios de administración (Profesor o estudiante) esta interfaz se mostrará, pero solo para brindar información.
- **Adicionar profesor:** En esta interfaz se podrá agregar al sistema un profesor que sea externo y que no este en la base de datos de la universidad, en caso de ser un usuario sin privilegios de administración (Profesor o estudiante) esta interfaz no se mostrará.
- **Eliminar profesor:** En esta interfaz se podrán eliminar de la base de datos de la aplicación los profesores que por una causa u otra son dados de baja de la universidad o se muevan a otra facultad, en caso de ser un usuario sin privilegios de administración (Profesor o estudiante) esta interfaz no se mostrará.
- **Lista de profesores:** En esta interfaz se muestra un listado de profesores con la carga que tiene cada uno, con un campo llamado operaciones donde se puede ver algunos datos del profesor y también se puede eliminar el mismo, esto último solo lo puede hacer el administrador del sistema, en caso de ser un usuario sin privilegios de administración (Profesor o estudiante) esta interfaz se mostrará, pero solo para brindar información.

- **Adicionar estudiante:** En esta interfaz se podrá adicionar un estudiante de otra facultad que esté realizando su trabajo de tesis en la facultad 8, este estudiante es adicionado a un grupo determinado, en caso de ser un usuario sin privilegios de administración (Profesor o estudiante) esta interfaz no se mostrará.
- **Eliminar estudiante:** En esta interfaz se podrán eliminar de la base de datos de la aplicación los estudiantes que por una causa u otra son dados de baja de la universidad o si hacen su tesis por otra facultad, en caso de ser un usuario sin privilegios de administración (Profesor o estudiante) esta interfaz no se mostrará.
- **Grupos:** En esta interfaz se muestra un listado de los grupos de quinto año y sus integrantes.
- **Sin tesis:** En esta interfaz se muestra un listado de los estudiantes, con sus grupos, a los que aún no se le ha asignado un tema de tesis.
- **Áreas de desarrollo:** En esta interfaz el jefe de comité de tesis puede crear o eliminar las diferentes áreas de desarrollo, en caso de ser un usuario sin privilegios de administración (Profesor o estudiante) esta interfaz se mostrará, pero solo para brindar información.

2.4.7 Requisitos no funcionales del sistema

Requerimientos de apariencia o interfaz externa

La aplicación propuesta será usada por personas que tengan o no conocimientos básicos de informática, la interfaz debe ser amigable y fácil de usar, de manera que no sea una dificultad para los usuarios el trabajo con la misma. La comunicación entre el servidor de base de datos y el servidor Web será mediante el protocolo TCP/IP, entre las máquinas clientes y el servidor Web será por HTTP y entre el servidor y el directorio activo mediante LDAP.

Requerimientos de usabilidad

A los administradores del sistema se les dará un adiestramiento básico en el uso de la aplicación. Estas personas tendrán un nivel de acceso amplio en la aplicación para poder darle respuesta a cada incidente ocurrido.

Requerimientos de rendimiento

Para un funcionamiento óptimo de la aplicación se seguirán las diferentes técnicas de elaboración de sitios Web, que faciliten el acceso rápido a sus páginas. La eficiencia del producto estará determinada en gran medida por el aprovechamiento de los recursos que se disponen en el modelo cliente/servidor,

y la velocidad de las consultas de la base de datos. La aplicación propuesta debe ser rápida y el tiempo de respuesta debe ser el mínimo posible, adecuado a la rapidez con que el cliente requiere la respuesta a su petición.

Requerimiento de portabilidad

Las herramientas utilizadas podrán ser usadas bajo cualquier sistema operativo Windows NT en adelante o cualquier distribución de Linux. El servidor Web y el servidor de Base de Datos pueden estar en la misma PC sin ocasionar problema alguno.

Requerimientos de seguridad

- **Confiablez:** La información manejada por el sistema debe estar protegida de acceso no autorizado.
- **Integridad:** La información manejada por el sistema debe ser objeto de cuidadosa protección contra la corrupción y estados de inconsistencia.
- **Disponibilidad:** La aplicación deberá estar disponible en todo momento para aquellas personas con acceso a la información y los mecanismos utilizados para lograr la seguridad no deben ser un obstáculo a los usuarios para obtener los datos deseados en un momento dado.

Requerimientos de software

En las computadoras de los usuarios solo se requiere un navegador Web, bajo cualquier sistema operativo Windows NT en adelante o cualquier distribución de Linux. En el servidor de base de datos se requiere Windows NT en adelante o cualquier distribución de Linux. Para su implementación se utilizará la herramienta de desarrollo NuSphere PhpED y como gestor de base de datos MySQL.

Requerimientos de hardware

En el cliente se requiere una máquina con 128 MB de RAM como mínimo, el servidor Web junto con el servidor de base de datos debe tener 256 MB de RAM y 20 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.

Restricción en el diseño y la implementación

Se debe realizar una aplicación que permita dar respuesta en el menor tiempo posible, garantizando la calidad del sistema apoyado completamente en el CMS Drupal. Para garantizar el desarrollo de la

aplicación se utilizará la metodología XP. Se utilizará para realizar los modelos UML y como herramienta de apoyo a este lenguaje de modelación se utiliza Rational Suite 2003.

Conclusiones

Teniendo en cuenta que en la facultad 8 no existe una aplicación de este tipo, se evidencia la necesidad de una aplicación dinámica que pueda ser visitada desde cualquier parte de la universidad y pueda brindar una amplia información a los usuarios necesitados. En este capítulo se realizó la descripción del objeto de estudio y automatización quedando demostrada la necesidad de la realización del sistema que se propone.

Capitulo III. Exploración y planificación.

3.1 Introducción.

En el presente capítulo se abordan las fases de exploración y planificación pertenecientes a la metodología de desarrollo XP (Extreme Programming) utilizada para la implementación del sistema que se propone. Además se exponen los artefactos generados durante el transcurso de dichas fases.

3.2 Fase de exploración.

La metodología de desarrollo *XP* (Extreme Programming) comienza con la fase de exploración. Durante esta se realiza el proceso de identificación de las historias de usuario, así como la familiarización de los equipos de trabajo con las tecnologías y herramientas seleccionadas para la construcción del proyecto. (Beck and Addison 2007)

3.2.1 Historias de Usuario.

Las historias de usuario son la forma en que se especifican en XP (Extreme Programming) los requisitos funcionales del sistema. Estas se escriben desde la perspectiva del cliente aunque los desarrolladores pueden brindar también su ayuda en la identificación de las mismas. El contenido de estas debe ser concreto y sencillo. Durante la fase de exploración se identificaron veinticuatro historias de usuario, las cuales se detallan a continuación.

Tabla 3.2. 1 Historia de usuario Autenticar Usuario

Historia de Usuario	
Número: 1	Usuario: Profesores, Estudiantes, Trabajadores.
Nombre historia: Autenticar usuario.	
Prioridad en negocio: Media	Riesgo en desarrollo: Bajo
Puntos estimados: 0.4	Iteración asignada: 1
Programador responsable: Emilio Salcerio Paz – Reynaldo Heredia Rodríguez	
Descripción:	

El usuario de la aplicación se autentica en la misma.
Observaciones:

Tabla 3.2. 2 Historia de usuario Eliminar usuario.

Historia de Usuario	
Número: 2	Usuario: Administrador del sistema
Nombre historia: Eliminar usuario.	
Prioridad en negocio: Media	Riesgo en desarrollo: Bajo
Puntos estimados: 0.4	Iteración asignada: 1
Programador responsable: Emilio Salcerio Paz – Reynaldo Heredia Rodríguez	
Descripción: El administrador del sistema elimina un usuario de la aplicación retirándole los privilegios en la misma en caso de que los posea.	
Observaciones:	

Tabla 3.2.3 Historia de usuario Crear de Comité de Tesis

Historia de Usuario	
Número: 3	Usuario: Jefe de comité de tesis
Nombre historia: Crear Comité de tesis	
Prioridad en negocio: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 1	Iteración asignada: 1
Programador responsable: Emilio Salcerio Paz – Reynaldo Heredia Rodríguez	
<p>Descripción:</p> <p>Se podrá gestionar los datos del Comité de tesis, crear la composición del Comité de tesis (profesores y jefes de proyecto).</p>	
Observaciones:	

Tabla 3.2.4 Historia de usuario Modificar de Comité de Tesis

Historia de Usuario	
Número: 4	Usuario: Jefe de comité de tesis
Nombre historia: Modificar Comité de tesis	
Prioridad en negocio: Alta	Riesgo en desarrollo: Media
Puntos estimados: 1	Iteración asignada: 1

Programador responsable: Emilio Salcerio Paz – Reynaldo Heredia Rodríguez
<p>Descripción:</p> <p>Se podrá gestionar los datos del Comité de tesis, se podrá modificar la composición del mismo (profesores y jefes de proyecto).</p>
Observaciones:

Tabla 3.2.5 Historia de usuario Eliminar de Comité de Tesis

Historia de Usuario	
Número: 5	Usuario: Jefe de comité de tesis
Nombre historia: Eliminar Comité de tesis	
Prioridad en negocio: Alta	Riesgo en desarrollo: Media
Puntos estimados: 0.3	Iteración asignada: 1
Programador responsable: Emilio Salcerio Paz – Reynaldo Heredia Rodríguez	
<p>Descripción:</p> <p>Se podrá gestionar los datos del Comité de tesis, se podrá eliminar la composición del mismo (profesores y jefes de proyecto) al término de sus funciones o por decisión de Jefe del comité de tesis.</p>	

Observaciones:

Tabla 3.2. 6 Historia de usuario Crear tribunal

Historia de Usuario	
Número: 6	Usuario: Jefe del Comité de tesis
Nombre historia: Crear Tribunal.	
Prioridad en negocio: Media	Riesgo en desarrollo: Alta
Puntos estimados: 1	Iteración asignada: 2
Programador responsable: Emilio Salcerio Paz – Reynaldo Heredia Rodríguez	
<p>Descripción:</p> <p>Se podrán Gestionar los datos del Tribunal, crear la composición del tribunal (están integrados por profesores. Dichos tribunales están conformados por, un presidente, vocales (uno o dos) y un secretario).</p>	
<p>Observaciones:</p> <p>Cada tribunal esta distribuido por perfiles.</p>	

Tabla 3.2. 7 Historia de usuario Modificar tribunal

Historia de Usuario	
Número: 7	Usuario: Jefe del Comité de tesis

Nombre historia: Modificar Tribunal.	
Prioridad en negocio: Media	Riesgo en desarrollo: Media
Puntos estimados: 0.3	Iteración asignada: 2
Programador responsable: Emilio Salcerio Paz – Reynaldo Heredia Rodríguez	
Descripción: Se podrán Gestionar los datos del Tribunal, modificar la composición del tribunal (Profesores que lo conforman) esta decisión será aprobada y ejecutada por el Jefe de Comité de tesis.	
Observaciones:	

Tabla 3.2. 8 Historia de usuario Eliminar tribunal

Historia de Usuario	
Número: 8	Usuario: Jefe del Comité de tesis
Nombre historia: Eliminar Tribunal.	
Prioridad en negocio: Media	Riesgo en desarrollo: Media
Puntos estimados: 0.3	Iteración asignada: 2
Programador responsable: Emilio Salcerio Paz – Reynaldo Heredia Rodríguez	
Descripción: Se podrán Gestionar los datos del Tribunal, eliminar la composición del tribunal (Profesores	

que lo conforman) esta decisión será aprobada y ejecutada por el Jefe de Comité de tesis.
Observaciones:

Tabla 3.2.9 Historia de usuario Publicar tema de tesis

Historia de Usuario	
Número: 9	Usuario: Profesor
Nombre historia: Publicar tema de tesis	
Prioridad en negocio: Media	Riesgo en desarrollo: Alta
Puntos estimados: 1	Iteración asignada: 3
Programador responsable: Emilio Salcerio Paz – Reynaldo Heredia Rodríguez	
<p>Descripción:</p> <p>El profesor es el encargado de publicar los temas de tesis que sea capaz de proponer, el profesor accede al sitio y publica el tema, de este modo el tema propuesto va a la base de datos como una propuesta de tema de tesis pendiente a ser aprobada.</p>	
Observaciones:	

Tabla 3.2.10 Historia de usuario Eliminar tema de tesis

Historia de Usuario

Número: 10	Usuario: Jefe de Comité de tesis
Nombre historia: Eliminar Tema de tesis.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Media
Puntos estimados: 0.3	Iteración asignada: 3
Programador responsable: Emilio Salcerio Paz – Reynaldo Heredia Rodríguez	
<p>Descripción:</p> <p>Se comprueba el contenido del tema de tesis en la pagina Web, se hace una revisión por parte del profesor y el comité de tesis en caso de que el tema propuesto no tenga la suficiente fuerza para ser un tema de tesis se comprueba si el tema está asignado a algún tesista en el campo “Tema asignado” en caso de que lo esté, es retirado y a los tesistas se le asigna otro tema de tesis de la base de datos, y el tema anterior es eliminado por el Jefe de comité de tesis.</p>	
Observaciones:	

Tabla 3.2. 11 Historia de usuario Eliminar tema de tesis

Historia de Usuario	
Número: 11	Usuario: Jefe de Comité de tesis
Nombre historia: Revisar Tema de tesis.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Media
Puntos estimados: 0.3	Iteración asignada: 3
Programador responsable: Emilio Salcerio Paz – Reynaldo Heredia Rodríguez	

<p>Descripción:</p> <p>El jefe de comité de tesis revisa el tema propuesto en caso de ser aprobado se le asigna un perfil en el caso contrario se le pone un resumen del por qué de su rechazo y pasa a la parte de los temas rechazados.</p>
<p>Observaciones:</p>

Tabla 3.2. 12 Historia de usuario Modificar tema de tesis

Historia de Usuario	
Número: 12	Usuario: Profesor
Nombre historia: Modificar Tema de Tesis	
Prioridad en negocio: Alta	Riesgo en desarrollo: Media
Puntos estimados: 1	Iteración asignada: 3
Programador responsable: Emilio Salcerio Paz – Reynaldo Heredia Rodríguez	
<p>Descripción:</p> <p>El Jefe de comité de tesis notifica al profesor que es necesario que modifique el tema de tesis para ser aprobado, en este caso el profesor puede modificar el tema de tesis que ya publicó, pero si este ya fue aprobado ya el profesor no puede ejercer ninguna acción sobre el mismo.</p>	
Observaciones:	

Tabla 3.2. 13 Historia de usuario Publicar cronograma

Historia de Usuario	
Número: 13	Usuario: Planificador
Nombre historia: Publicar cronograma.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alta
Puntos estimados: 1	Iteración asignada: 3
Programador responsable: Emilio Salcerio Paz – Reynaldo Heredia Rodríguez	
<p>Descripción:</p> <p>Es publicado el cronograma para lo tribunales en los cortes de tesis.</p>	
<p>Observaciones:</p>	

Tabla 3.2. 14 Historia de usuario Asignar tema de tesis

Historia de Usuario	
Número: 14	Usuario: Jefe de Comité de tesis.
Nombre historia: Asignar Tema de Tesis	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alta
Puntos estimados: 1	Iteración asignada: 3
Programador responsable: Emilio Salcerio Paz – Reynaldo Heredia Rodríguez	

<p>Descripción:</p> <p>Al seleccionar asignar tema de tesis se muestra una plantilla con campos a llenar los mismos son: tutor, desarrolladores de la tesis, que serán los estudiantes encargados de desarrollar esta tema de tesis; al aceptar el tema de tesis con sus datos es actualizado en la base de datos pasando al estado de asignado, esta acción es desarrollada por el Jefe de comité de tesis.</p>
<p>Observaciones:</p>

Tabla 3.2. 15 Historia de usuario Evaluar Tema de Tesis.

Historia de Usuario	
Número: 15	Usuario: Jefe de Tribunal
Nombre historia: Evaluar Trabajo de Tesis	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alta
Puntos estimados: 0.4	Iteración asignada: 3
Programador responsable: Emilio Salcerio Paz – Reynaldo Heredia Rodríguez	
<p>Descripción:</p> <p>Esta acción será realizada por el Jefe de tribunal correspondiente a la tesis evaluada, luego de un corte de tesis el Jefe de tribunal actualizara el seguimiento de dicha tesis con la nota alcanzada y recomendaciones en caso de que las tenga.</p>	

Observaciones:

Tabla 3.2. 16 Historia de usuario Listar profesores

Historia de Usuario	
Número: 16	Usuario: Profesores
Nombre historia: Listar profesores.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alta
Puntos estimados: 0.4	Iteración asignada: 3
Programador responsable: Emilio Salcerio Paz – Reynaldo Heredia Rodríguez	
<p>Descripción:</p> <p>Muestra un listado de profesores con su carga (Cantidad de roles que desempeña el profesor en el proceso de tesis) y en el caso del administrador le muestra la opción de eliminar el mismo.</p>	
Observaciones:	

Tabla 3.2. 17 Historia de usuario Listar grupos de quinto año

Historia de Usuario

Número: 17	Usuario: Estudiantes y profesores.
Nombre historia: Listar grupos de quinto año.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alta
Puntos estimados: 0.4	Iteración asignada: 3
Programador responsable: Emilio Salcerio Paz – Reynaldo Heredia Rodríguez	
Descripción: Muestra un listado de los grupos de estudiantes de quinto año	
Observaciones:	

Tabla 3.2. 18 Historia de usuario Listar estudiantes sin tesis.

Historia de Usuario	
Número: 18	Usuario: Profesores y estudiantes.
Nombre historia: Listar estudiantes sin tesis.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alta
Puntos estimados: 0.4	Iteración asignada: 3
Programador responsable: Emilio Salcerio Paz – Reynaldo Heredia Rodríguez	
Descripción: Se muestra un listado de los estudiantes de quinto año que no tengan un tema de tesis asignado.	

Observaciones:

Tabla 3.2. 19 Historia de usuario Adicionar estudiantes.

Historia de Usuario	
Número: 19	Usuario: Jefe de comité de tesis
Nombre historia: Adicionar estudiantes.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alta
Puntos estimados: 0.4	Iteración asignada: 3
Programador responsable: Emilio Salcerio Paz – Reynaldo Heredia Rodríguez	
<p>Descripción:</p> <p>Se muestra un listado de los estudiantes de quinto año ordenado por sus grupos, debajo aparece un botón adicionar estudiante donde debe de introducir su nombre y apellido.</p>	
<p>Observaciones:</p>	

Tabla 3.2. 20 Historia de usuario Eliminar estudiantes.

Historia de Usuario

Número: 20	Usuario: Jefe de comité de tesis
Nombre historia: Eliminar estudiantes.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alta
Puntos estimados: 0.4	Iteración asignada: 3
Programador responsable: Emilio Salcerio Paz – Reynaldo Heredia Rodríguez	
<p>Descripción:</p> <p>Se muestra un listado de los estudiantes de quinto año ordenado por sus grupos, al lado de cada estudiante la opción de eliminar el mismo de la base de datos de la aplicación.</p>	
Observaciones:	

Tabla 3.2. 21 Historia de usuario Adicionar profesores.

Historia de Usuario	
Número: 21	Usuario: Jefe de comité de tesis
Nombre historia: Adicionar profesores.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alta
Puntos estimados: 0.4	Iteración asignada: 3
Programador responsable: Emilio Salcerio Paz – Reynaldo Heredia Rodríguez	
<p>Descripción:</p> <p>Se muestra un listado de los profesores, debajo aparece un botón adicionar profesor donde</p>	

debe de introducir su nombre y apellido y si es docente o no.
Observaciones:

Tabla 3.2. 22 Historia de usuario Eliminar profesores.

Historia de Usuario	
Número: 22	Usuario: Jefe de comité de tesis
Nombre historia: Eliminar profesores.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alta
Puntos estimados: 0.4	Iteración asignada: 3
Programador responsable: Emilio Salcerio Paz – Reynaldo Heredia Rodríguez	
<p>Descripción:</p> <p>Se muestra un listado de los profesores al lado de cada profesor su carga y la opción de eliminar el mismo de la base de datos de la aplicación.</p>	
Observaciones:	

Tabla 3.2. 23 Historia de usuario Modificar la evaluación del trabajo de tesis.

Historia de Usuario	
Número: 23	Usuario: Jefe de comité de tesis

Nombre historia: Modificar la evaluación del trabajo de tesis.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alta
Puntos estimados: 0.4	Iteración asignada: 3
Programador responsable: Emilio Salcerio Paz – Reynaldo Heredia Rodríguez	
Descripción: Para cada evaluación de tesis existente hay una opción de modificar la misma en caso de que exista una equivocación en la nota	
Observaciones:	

Tabla 3.2. 24 Historia de usuario Eliminar evaluaciones.

Historia de Usuario	
Número: 24	Usuario: Jefe de comité de tesis
Nombre historia: Eliminar evaluaciones.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alta
Puntos estimados: 0.4	Iteración asignada: 3
Programador responsable: Emilio Salcerio Paz – Reynaldo Heredia Rodríguez	
Descripción: Para cada evaluación de tesis existente hay una opción de eliminar la misma en caso de que exista una equivocación en la nota o por decisión de Jefe de comité de tesis.	

Observaciones:

3.3 Planificación

Durante la fase de planificación se realiza una estimación del esfuerzo que costará implementar cada historia de usuario. Este se expresa utilizado como medida el punto. Un punto se considera como una semana ideal de trabajo donde los miembros de los equipos de desarrollo trabajan el tiempo planeado sin ningún tipo de interrupción. Esta estimación incluye todo el esfuerzo asociado a la implementación de la historia de usuario.

3.3.1. Estimación de esfuerzo por Historias de Usuario

Para el desarrollo de la aplicación propuesta en este trabajo se realizó una estimación del esfuerzo para cada una de las historias de usuario identificadas, llegándose a los resultados que se muestran en la siguiente tabla.

Tabla 3.3.1 Estimación de esfuerzo por historia de usuario.

Historias de Usuario	Puntos estimados
Autenticar usuario	0.4
Eliminar usuario	0.4
Crear Comité de tesis	1
Modificar Comité de tesis	1
Eliminar Comité de tesis	0.3
Cronograma	0.5
Crear Tribunal.	1
Modificar tribunal	0.3
Eliminar tribunal	0.3
Publicar tema de tesis	1
Eliminar Tema de tesis.	0.3
Modificar tema de tesis	1
Aprobar Tema de tesis.	1
Asignar Tema de Tesis	1

Evaluar Trabajo de Tesis	0.4
Modificar la evaluación de los trabajos de tesis	0.5
Eliminar evaluaciones	0.4
Listar profesores	0.4
Adicionar profesores	0.6
Eliminar profesores	0.6
Listar grupos de quinto año	0.4
Adicionar estudiantes	0.6
Eliminar estudiantes	0.6
Listar estudiantes sin tesis	0.4

3.3.2. Plan de Iteraciones.

Una vez identificadas las historias de usuario del sistema y estimado el esfuerzo dedicado a la realización de cada una de estas se procede a la planificación de la etapa de implementación de la aplicación. Se decidió realizar la implementación de la aplicación en tres iteraciones, detalladas a continuación.

3.3.3. Iteración 1

En esta iteración se realizarán las historias de usuarios elegidas para este primer módulo, la realización de las mismas va dando una idea de cómo quedara la aplicación aunque todavía estará en sus inicios.

3.3.4. Iteración 2

La implementación de estas historias de usuarios en esta iteración da una idea más completa de la aplicación en cuestión.

3.3.5. Iteración 3.

La implementación de estas historias de usuarios en esta iteración da fin a la implementación de las funcionalidades más críticas para la aplicación.

3.3.6. Iteración 4

La implementación de estas historias de usuarios en esta iteración da fin a la implementación de la aplicación.

3.3.6 Plan de duración de las iteraciones.

Como parte del ciclo de vida de un proyecto utilizando la metodología XP se crea el plan de duración de cada una de las iteraciones, según los equipos de desarrollo con que se cuente, en este caso se hace para el único equipo de desarrollo que se tiene. Este plan se encarga de mostrar las historias de usuario que serán cumplimentadas en cada una de las iteraciones, así como la duración estimada de de cada una y el orden en que se implementarán.

Tabla 3.3.2 Equipo de desarrollo1

Iteración	Orden de las historias de usuario a implementar	Duración total de la iteración
Iteración 1	1.1- Autenticar usuario. 1.2- Eliminar usuario. 1.3- Crear Comité de tesis. 1.4- Modificar Comité de tesis. 1.5- Eliminar Comité de tesis.	4.5 semana.
Iteración 2	2.1- Cronograma. 2.2- Crear Tribunal. 2.3- Modificar tribunal. 2.4- Eliminar tribunal.	2.5 semanas.
Iteración 3	3.1- Publicar tema de tesis. 3.2- Eliminar Tema de tesis. 3.3- Modificar tema de tesis. 3.4- Aprobar Tema de tesis. 3.5- Asignar Tema de Tesis. 3.6- Evaluar Trabajo de Tesis. 3.7- Modificar la evaluación de los trabajos de tesis. 3.8- Eliminar evaluaciones	5 semanas.
	4.1- Listar profesores.	

Iteración 4	4.2- Adicionar profesores. 4.3- Eliminar profesores. 4.4- Listar grupos de quinto año. 4.5- Adicionar estudiantes. 4.6- Eliminar estudiantes. 4.7- Listar estudiantes sin tesis.	2 semanas.
-------------	---	------------

3.3.7 Plan de entregas

Plan de entregas ideado para la fase de implementación. Como producto del mismo se harán *releases* del sistema al finalizar cada iteración en la fecha aproximada que se indica en la siguiente tabla.

Tabla 3.3. 3 Plan de entregas

Módulo	Final 1ra Iteración 4ta semana de febrero.	Final 2da Iteración 4ta semana de marzo.	Final 3ra Iteración 1ra semana de mayo	Final de la cuarta iteración 3ra semana de mayo
Calidad	0.1	0.6	0.9	0.1

Conclusiones

Como cuerpo de este capítulo se abordó todo lo referente a las fases de exploración y planificación del proyecto, haciendo una descripción de cada uno de los artefactos generados en las mismas durante su desarrollo dando una idea de la magnitud del proyecto a realizar.

Capitulo IV. Implementación y Pruebas

4.1 Introducción.

XP plantea que la implementación de un software debe realizarse de forma iterativa, obteniendo al culminar cada iteración un producto funcional que debe ser probado y mostrado al cliente para retroalimentar a los desarrolladores con su opinión. En el presente capítulo se detallan las cuatro iteraciones llevadas a cabo durante la etapa de construcción del sistema, exponiendo las tareas

generadas por cada historia de usuario, así como las pruebas de aceptación efectuadas sobre el proyecto.

Durante el transcurso de las iteraciones se realiza la implementación de las historias de usuario seleccionadas para cada una de estas. Al inicio de las mismas, se lleva a cabo una revisión del plan de iteraciones y se modifica de ser necesario. Ajustándose a la planificación realizada, se llevaron a cabo cuatro iteraciones de desarrollo sobre el sistema, obteniéndose al finalizar, un producto listo para su puesta en función. A continuación se detalla cada una de las iteraciones. (Fernández Escribano 2007)

4.2 Iteración 1

Durante esta iteración se abordaron las historias de usuario elegidas para el primer módulo y se construyó la base de la arquitectura del sistema con el fin de obtener un producto con las funcionalidades primarias para ser mostrado al cliente y obtener una rápida y amplia retroalimentación de este.

Tabla 4.2 1 Historias de usuario implementadas en la primera iteración.

Historias de Usuario	Estimación	Real
Autenticar usuario.	0.4	0.5
Eliminar usuario.	0.4	0.5
Crear Comité de tesis	1	1
Modificar Comité de tesis.	1	1
Eliminar Comité de tesis.	0.3	0.5

4.2.1 Tareas de las historias de usuario implementadas en la primera iteración.

Autenticar Usuario.

Tabla 4.2 2 Tarea #1 de la historia de usuario Autenticar Usuario.

Tarea

Número tarea: 1	Número historia: 1
Nombre tarea: Configuración de la interfaz para la autenticación de usuarios	
Tipo de tarea : Desarrollo	Puntos estimados: 0.075
Fecha inicio: 1 de Febrero 2008	Fecha fin: 3 de Febrero 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
Descripción: Se configura la interfaz para la autenticación de los usuarios que harán uso de la aplicación.	

Tabla 4.2 3 Tarea #2 de la historia de usuario Autenticar Usuario.

Tarea	
Número tarea: 2	Número historia: 1
Nombre tarea: Comprobar si el usuario es válido	
Tipo de tarea : Desarrollo	Puntos estimados: 0.1
Fecha inicio: 4 de Febrero 2008	Fecha fin: 6 de Febrero 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
Descripción: Se comprueban los datos del usuario autenticado verificando si es válido o no.	

Tabla 4.2 4 Tarea #3 de la historia de usuario Autenticar Usuario.

Tarea	
Número tarea: 3	Número historia: 1
Nombre tarea: Acceso con los correspondientes permisos.	
Tipo de tarea : Desarrollo	Puntos estimados: 0.1
Fecha inicio: 6 de Febrero 2008	Fecha fin: 8 de Febrero 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
Descripción: Se le concede permiso de acceder a la aplicación si el usuario es válido con los permisos correspondientes al mismo.	

Eliminar usuario.

Tabla 4.2 5 Tarea #1 de la historia de usuario Eliminar Usuario.

Tarea	
Número tarea: 1	Número historia: 2
Nombre tarea: Configuración de la interfaz para eliminar un usuario	
Tipo de tarea : Desarrollo	Puntos estimados: 0.1
Fecha inicio: 8 de Febrero 2008	Fecha fin: 9 de Febrero 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
Descripción: Se configura la interfaz para la eliminación de los privilegios del usuario.	

Tabla 4.2 6 Tarea #2 de la historia de usuario Eliminar Usuario.

Tarea	
Número tarea: 2	Número historia: 2
Nombre tarea: Verificar listado de usuarios y sus privilegios para la selección del usuario a eliminar	
Tipo de tarea : Desarrollo	Puntos estimados: 0.1
Fecha inicio: 9 de Febrero 2008	Fecha fin: 10 de Febrero 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
Descripción: Se muestra un listado de los usuarios con privilegios en la aplicación para seleccionar el que será eliminado.	

Tabla 4.2 7 Tarea #3 de la historia de usuario Eliminar Usuario.

Tarea	
Número tarea: 3	Número historia: 2
Nombre tarea: Eliminar privilegios del usuario de la base de datos	
Tipo de tarea : Desarrollo	Puntos estimados: 0.1
Fecha inicio: 10 de Febrero 2008	Fecha fin: 11 de Febrero 2008

Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez
Descripción: Los datos y privilegios del usuario seleccionado son eliminados de la base de datos.

Crear Comité de tesis

Tabla 4.2 8 Tarea #1 de la historia de usuario Crear Comité de tesis.

Tarea	
Número tarea: 1	Número historia: 3
Nombre tarea: Creación de la interfaz	
Tipo de tarea : Desarrollo	Puntos estimados: 0.33
Fecha inicio: 12 de Febrero 2008	Fecha fin: 13 de Febrero 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
Descripción: Se crea la interfaz para Crear Comité de tesis.	

Tabla 4.2 9 Tarea #2 de la historia de usuario Crear Comité de tesis.

Tarea	
Número tarea: 2	Número historia: 3
Nombre tarea: Inserción de los datos referentes para la creación del Comité de tesis	
Tipo de tarea : Desarrollo	Puntos estimados: 0.33
Fecha inicio: 13 de Febrero 2008	Fecha fin: 15 de Febrero 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
Descripción: Se insertan los datos necesarios para crear el Comité de tesis.	

Tabla 4.2 10 Tarea #3 de la historia de usuario Crear Comité de tesis.

Tarea	
Número tarea: 3	Número historia: 3
Nombre tarea: Adición de los datos en la base de datos	
Tipo de tarea : Desarrollo	Puntos estimados: 0.33
Fecha inicio: 15 de Febrero 2008	Fecha fin: 16 de Febrero 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
Descripción: Se adicionan los datos del comité de tesis en la base de datos y se actualiza la base de datos.	

Modificar Comité de tesis.

Tabla 4.2 11 Tarea #1 de la historia de usuario Modificar Comité de tesis.

Tarea	
Número tarea: 1	Número historia: 4
Nombre tarea: Creación de la interfaz	
Tipo de tarea : Desarrollo	Puntos estimados: 0.33
Fecha inicio: 16 de Febrero 2008	Fecha fin: 17 de Febrero 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
Descripción: Se crea la interfaz para Modificar Comité de tesis.	

Tabla 4.2 12 Tarea #2 de la historia de usuario Modificar Comité de tesis.

Tarea	
Número tarea: 2	Número historia: 4
Nombre tarea: Modificación de los datos del comité de tesis	
Tipo de tarea : Desarrollo	Puntos estimados: 0.33
Fecha inicio: 17 de Febrero 2008	Fecha fin: 20 de Febrero 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	

<p>Descripción: Se modifican los datos correspondientes en el comité de tesis.</p>

Tabla 4.2 13 Tarea #3 de la historia de usuario Modificar Comité de tesis.

Tarea	
Número tarea: 3	Número historia: 4
Nombre tarea: Adición de los datos en la base de datos	
Tipo de tarea : Desarrollo	Puntos estimados: 0.33
Fecha inicio: 20 de Febrero 2008	Fecha fin: 22 de Febrero 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
<p>Descripción: Se adicionan los datos del comité de tesis en la base de datos y se actualiza la base de datos.</p>	

Eliminar Comité de tesis.

Tabla 4.2 14 Tarea #1 de la historia de usuario Eliminar Comité de tesis.

Tarea	
Número tarea: 1	Número historia: 5
Nombre tarea: Creación de la interfaz	
Tipo de tarea : Desarrollo	Puntos estimados: 0.33
Fecha inicio: 22 de Febrero 2008	Fecha fin: 24 de Febrero 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
<p>Descripción: Se crea la interfaz para eliminar los datos del Comité de tesis.</p>	

Tabla 4.2 15 Tarea #2 de la historia de usuario Eliminar Comité de tesis.

Tarea

Número tarea: 2	Número historia: 5
Nombre tarea: Selección de los datos a eliminar.	
Tipo de tarea : Desarrollo	Puntos estimados: 0.33
Fecha inicio: 24 de Febrero 2008	Fecha fin: 26 de Febrero 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
Descripción: Se seleccionan los datos del Comité de tesis para posteriormente ser eliminados.	

Tabla 4.2 16 Tarea #3 Historia de usuario Eliminar Comité de tesis.

Tarea	
Número tarea: 3	Número historia: 5
Nombre tarea: Eliminación de los datos seleccionados de la base de datos.	
Tipo de tarea : Desarrollo	Puntos estimados: 0.33
Fecha inicio: 26 de Febrero 2008	Fecha fin: 27 de Febrero 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
Descripción: Se eliminan los datos seleccionados de la base de datos y se actualiza la misma.	

4.3 Iteración 2

En esta iteración se implementan las funcionalidades del segundo módulo de la aplicación en esta fase se percibirá una idea mas amplia de cómo quedará la funcionalidad de la aplicación al final de su desarrollo.

Tabla 4.3. 1 Historias de usuario implementadas en la segunda iteración.

Historias de Usuario	Estimación	Real
Eliminar tribunal	0.3	0.5
Crear Tribunal.	1	1
Modificar tribunal.	0.3	0.5

Cronograma	0.5	0.5
------------	-----	-----

4.3.1 Tareas de las historias de usuario implementadas en la segunda iteración.

Crear Tribunal.

Tabla 4.3.2 Tarea #1 Historia de usuario Crear tribunal.

Tarea	
Número tarea: 1	Número historia: 6
Nombre tarea: Creación de la interfaz	
Tipo de tarea : Desarrollo	Puntos estimados: 0.33
Fecha inicio: 1 de marzo 2008	Fecha fin: 2 de marzo 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
Descripción: Se crea la interfaz para Crear tribunal.	

Tabla 4.3.3 Tarea #2 Historia de usuario Crear tribunal.

Tarea	
Número tarea: 2	Número historia: 6
Nombre tarea: Inserción de los datos referentes para la creación del tribunal	
Tipo de tarea : Desarrollo	Puntos estimados: 0.33
Fecha inicio: 4 de marzo 2008	Fecha fin: 6 de marzo 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
Descripción: Se insertan los datos necesarios para crear el tribunal.	

Tabla 4.3.4 Tarea #3 Historia de usuario Crear tribunal.

Tarea	
Número tarea: 3	Número historia: 6
Nombre tarea: Adición de los datos en la base de datos	
Tipo de tarea : Desarrollo	Puntos estimados: 0.33

Fecha inicio: 6 de marzo 2008	Fecha fin: 8 de marzo 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
Descripción: Se adicionan los datos del tribunal en la base de datos y se actualiza la misma.	

Modificar Tribunal.

Tabla 4.3. 5 Tarea #1 Historia de usuario Modificar tribunal.

Tarea	
Número tarea: 1	Número historia: 7
Nombre tarea: Creación de la interfaz	
Tipo de tarea : Desarrollo	Puntos estimados: 0.33
Fecha inicio: 8 de marzo 2008	Fecha fin: 10 de marzo 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
Descripción: Se crea la interfaz para Modificar los datos del tribunal.	

Tabla 4.3. 6 Tarea #2 Historia de usuario Modificar tribunal.

Tarea	
Número tarea: 2	Número historia: 7
Nombre tarea: Selección del tribunal a modificar.	
Tipo de tarea : Desarrollo	Puntos estimados: 0.33
Fecha inicio: 10 de marzo 2008	Fecha fin: 12 de marzo 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
Descripción: Se selecciona el tribunal a modificar de una lista de tribunales	

--

Tabla 4.3.7 Tarea #3 Historia de usuario Modificar tribunal.

Tarea	
Número tarea: 3	Número historia: 7
Nombre tarea: Modificación de los datos del tribunal	
Tipo de tarea : Desarrollo	Puntos estimados: 0.33
Fecha inicio: 12 de marzo 2008	Fecha fin: 13 de marzo 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
Descripción: Se modifican los datos correspondientes al tribunal seleccionado.	

Tabla 4.3.8 Tarea #4 Historia de usuario Modificar tribunal.

Tarea	
Número tarea: 4	Número historia: 7
Nombre tarea: Adición de los datos del tribunal en la base de datos	
Tipo de tarea : Desarrollo	Puntos estimados: 0.33
Fecha inicio: 13 de marzo 2008	Fecha fin: 15 de marzo 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
Descripción: Se adicionan los datos del tribunal en la base de datos y se actualiza la misma.	

Eliminar Tribunal.

Tabla 4.3.9 Tarea #1 Historia de usuario Eliminar tribunal.

Tarea	
Número tarea: 1	Número historia: 8
Nombre tarea: Creación de la interfaz	

Tipo de tarea : Desarrollo	Puntos estimados: 0.33
Fecha inicio: 15 de marzo 2008	Fecha fin: 16 de marzo 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
Descripción: Se crea la interfaz para Eliminar los datos del tribunal.	

Tabla 4.3. 10 Tarea #2 Historia de usuario Eliminar tribunal.

Tarea	
Número tarea: 2	Número historia: 8
Nombre tarea: Selección del tribunal a eliminar.	
Tipo de tarea : Desarrollo	Puntos estimados: 0.33
Fecha inicio: 16 de marzo 2008	Fecha fin: 18 de marzo 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
Descripción: Se selecciona el tribunal a eliminar de una lista de tribunales.	

Tabla 4.3. 11 Tarea #3 Historia de usuario Eliminar tribunal.

Tarea	
Número tarea: 3	Número historia: 8
Nombre tarea: Eliminación de los datos del tribunal seleccionado en la base de datos	
Tipo de tarea : Desarrollo	Puntos estimados: 0.33
Fecha inicio: 18 de marzo 2008	Fecha fin: 20 de marzo 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
Descripción: Se eliminan los datos del tribunal en la base de datos y se actualiza la misma.	

Cronograma

Tabla 4.3. 12 Tarea #1 Historia de usuario Cronograma.

Tarea

Número tarea: 1	Número historia: 9
Nombre tarea: Creación de la interfaz	
Tipo de tarea : Desarrollo	Puntos estimados: 0.33
Fecha inicio: 20 de marzo 2008	Fecha fin: 23 de marzo 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
Descripción: Se crea la interfaz para editar los datos del cronograma.	

Tabla 4.3. 13 Tarea #2 Historia de usuario Cronograma.

Tarea	
Número tarea: 2	Número historia: 9
Nombre tarea: Adición de los datos del cronograma en la base de datos	
Tipo de tarea : Desarrollo	Puntos estimados: 0.33
Fecha inicio: 23 de marzo 2008	Fecha fin: 27 de marzo 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
Descripción: Se llenan los campos necesarios y se adicionan los datos del cronograma en la base de datos y se actualiza la misma.	

4.4 Iteración 3

En el transcurso de esta iteración se implementaron las historias de usuario elegidas para el tercer módulo. Al culminar esta, se consta de un producto casi listo para su puesta en funcionamiento con la mayoría de sus funcionalidades más críticas ya implementadas.

Tabla 4.4. 1 Historias de usuarios implementadas en la tercera iteración.

Historias de Usuario	Estimación	Real
Publicar tema de tesis.	1	1

Eliminar Tema de tesis.	0.3	0.3
Modificar tema de tesis	1	1
Aprobar Tema de tesis.	1	1
Asignar Tema de Tesis.	1	1
Evaluar Trabajo de Tesis.	0.4	0.4
Modificar la evaluación de los trabajos de tesis.	0.5	0.5
Eliminar evaluaciones.	0.4	0.4

4.4.2 Tareas de las historias de usuario implementadas en la tercera iteración.

Publicar tema de tesis.

Tabla 4.4. 2 Tarea #1 Historia de usuario Publicar tema de tesis.

Tarea	
Número tarea: 1	Número historia: 10
Nombre tarea: Creación de la interfaz	
Tipo de tarea : Desarrollo	Puntos estimados: 0.33
Fecha inicio: 1 de abril 2008	Fecha fin: 3 de abril 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
Descripción: Se crea la interfaz para publicar los datos de un tema de tesis propuesto.	

Tabla 4.4. 3 Tarea #2 Historia de usuario Publicar tema de tesis.

Tarea	
Número tarea: 2	Número historia: 10
Nombre tarea: Adición de los datos del tribunal en la base de datos	
Tipo de tarea : Desarrollo	Puntos estimados: 0.33
Fecha inicio: 3 de abril 2008	Fecha fin: 5 de abril 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
Descripción: Se adicionan los datos del tribunal en la base de datos y se actualiza la misma.	

Eliminar Tema de tesis.

Tabla 4.4. 4 Tarea #1 Historia de usuario Eliminar tema de tesis.

Tarea	
Número tarea: 1	Número historia: 11
Nombre tarea: Creación de la interfaz	
Tipo de tarea : Desarrollo	Puntos estimados: 0.33
Fecha inicio: 5 de abril 2008	Fecha fin: 7 de abril 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
Descripción: Se crea la interfaz para eliminar los datos de un tema de tesis propuesto.	

Tabla 4.4. 5 Tarea #2 Historia de usuario Eliminar tema de tesis.

Tarea	
Número tarea: 2	Número historia: 11
Nombre tarea: Selección del tema de tesis a eliminar.	
Tipo de tarea : Desarrollo	Puntos estimados: 0.33
Fecha inicio: 6 de abril 2008	Fecha fin: 8 de abril 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
Descripción: Se selecciona el tema de tesis a eliminar de una lista de temas.	

Tabla 4.4. 6 Tarea #3 Historia de usuario Eliminar tema de tesis.

Tarea	
Número tarea: 2	Número historia: 11
Nombre tarea: Eliminación de los datos del tema de tesis en la base de datos	
Tipo de tarea : Desarrollo	Puntos estimados: 0.33

Fecha inicio: 8 de abril 2008	Fecha fin: 10 de abril 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
Descripción: Se eliminan los datos del tema de tesis en la base de datos y se actualiza la misma.	

Modificar tema de tesis.

Tabla 4.4.7 Tarea #1 Historia de usuario Modificar tema de tesis.

Tarea	
Número tarea: 1	Número historia: 12
Nombre tarea: Creación de la interfaz	
Tipo de tarea : Desarrollo	Puntos estimados: 0.33
Fecha inicio: 10 de abril 2008	Fecha fin: 11 de abril 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
Descripción: Se crea la interfaz para modificar los datos de un tema de tesis propuesto.	

Tabla 4.4.8 Tarea #2 Historia de usuario Modificar tema de tesis.

Tarea	
Número tarea: 2	Número historia: 12
Nombre tarea: Selección del tema de tesis a modificar.	
Tipo de tarea : Desarrollo	Puntos estimados: 0.33
Fecha inicio: 12 de abril 2008	Fecha fin: 14 de abril 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
Descripción: Se selecciona el tema de tesis a modificar de una lista de temas.	
Observación: Una vez que el tema de tesis esta aprobado no se puede realizar ninguna modificación sobre el mismo.	

--

Tabla 4.4.9 Tarea #3 Historia de usuario Modificar tema de tesis.

Tarea	
Número tarea: 3	Número historia: 12
Nombre tarea: Modificación de los datos del tema de tesis en la base de datos	
Tipo de tarea : Desarrollo	Puntos estimados: 0.33
Fecha inicio: 14 de abril 2008	Fecha fin: 15 de abril 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
Descripción: Se modifican los datos del tema de tesis en la base de datos y se actualiza la misma.	

Aprobar Tema de tesis.

Tabla 4.4.10 Tarea #1 Historia de usuario Aprobar tema de tesis.

Tarea	
Número tarea: 1	Número historia: 13
Nombre tarea: Creación de la interfaz	
Tipo de tarea : Desarrollo	Puntos estimados: 0.33
Fecha inicio: 15 de abril 2008	Fecha fin: 16 de abril 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
Descripción: Se crea la interfaz para aprobar un tema de tesis.	

Tabla 4.4.11 Tarea #2 Historia de usuario Aprobar tema de tesis.

Tarea	
Número tarea: 2	Número historia: 13
Nombre tarea: Selección del tema de tesis a aprobar.	
Tipo de tarea : Desarrollo	Puntos estimados: 0.33

Fecha inicio: 16 de abril 2008	Fecha fin: 18 de abril 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
Descripción: Se selecciona el tema de tesis a revisar de una lista de temas.	

Tabla 4.4. 12 Tarea #3 Historia de usuario Aprobar tema de tesis.

Tarea	
Número tarea: 3	Número historia: 13
Nombre tarea: Aprobar tema de tesis.	
Tipo de tarea : Desarrollo	Puntos estimados: 0.33
Fecha inicio: 18 de abril 2008	Fecha fin: 20 de abril 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
Descripción: Luego de haber estudiado el tema si se decide aprobarlo se llenan los campos necesarios y se pincha en el botón aceptar.	

Tabla 4.4. 13 Tarea #4 Historia de usuario Aprobar tema de tesis.

Tarea	
Número tarea: 3	Número historia: 13
Nombre tarea: Modificación de los datos del tema de tesis en la base de datos	
Tipo de tarea : Desarrollo	Puntos estimados: 0.33
Fecha inicio: 20 de de abril 2008	Fecha fin: 22 de abril 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
Descripción: Se modifican los datos del tema de tesis en la base de datos pasando a propuesto al campo de tesis libres y se actualiza la base de datos.	

Asignar Tema de tesis.

Tabla 4.4. 14 Tarea #1 Historia de usuario Asignar tema de tesis.

Tarea	
Número tarea: 1	Número historia: 14
Nombre tarea: Creación de la interfaz	
Tipo de tarea : Desarrollo	Puntos estimados: 0.33
Fecha inicio: 22 de abril 2008	Fecha fin: 24 de abril 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
Descripción: Se crea la interfaz para asignar un tema de tesis.	

Tabla 4.4. 15 Tarea #2 Historia de usuario Asignar tema de tesis.

Tarea	
Número tarea: 2	Número historia: 14
Nombre tarea: Elección del tema de tesis	
Tipo de tarea : Desarrollo	Puntos estimados: 0.33
Fecha inicio: 24 de abril 2008	Fecha fin: 26 de abril 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
Descripción: Se elige el tema de tesis de una lista de temas libres.	

Tabla 4.4. 16 Tarea #3 Historia de usuario Asignar tema de tesis.

Tarea	
Número tarea: 3	Número historia: 14
Nombre tarea: Asignación del tema de tesis elegido	
Tipo de tarea : Desarrollo	Puntos estimados: 0.33
Fecha inicio: 26 de abril 2008	Fecha fin: 28 de abril 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
Descripción: Se llenan los campos necesarios se adiciona el tema de tesis asignado a la base de	

datos y se actualiza la misma.

Evaluar Trabajo de Tesis.

Tabla 4.4. 17 Tarea #1 Historia de usuario Evaluar trabajo de tesis.

Tarea	
Número tarea: 1	Número historia: 15
Nombre tarea: Creación de la interfaz	
Tipo de tarea : Desarrollo	Puntos estimados: 0.33
Fecha inicio: 28 de abril 2008	Fecha fin: 29 de abril 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
Descripción: Se crea la interfaz para evaluar un trabajo de tesis.	

Tabla 4.4. 18 Tarea #2 Historia de usuario Evaluar trabajo de tesis.

Tarea	
Número tarea: 2	Número historia: 15
Nombre tarea: Selección de la tesis a evaluar	
Tipo de tarea : Desarrollo	Puntos estimados: 0.33
Fecha inicio: 29 de abril 2008	Fecha fin: 30 de abril 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
Descripción: Se selecciona según el corte que sea de un listado de cortes que dentro tendrán un listado de las tesis.	

Tabla 4.4. 19 Tarea #3 Historia de usuario Evaluar trabajo de tesis.

Tarea	
Número tarea: 3	Número historia: 15
Nombre tarea: Selección de la tesis a evaluar	
Tipo de tarea : Desarrollo	Puntos estimados: 0.33

Fecha inicio: 30 de abril 2008	Fecha fin: 1 de mayo 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
Descripción: Se selecciona según el corte que sea de un listado de cortes que dentro tendrán un listado de las tesis.	

Tabla 4.4. 20 Tarea #4 Historia de usuario Evaluar trabajo de tesis.

Tarea	
Número tarea: 4	Número historia: 15
Nombre tarea: Actualización de la base de datos	
Tipo de tarea : Desarrollo	Puntos estimados: 0.33
Fecha inicio: 1 de mayo 2008	Fecha fin: 2 de mayo 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
Descripción: Después de haber llenado los campos necesarios se agrega a la base de datos y se actualiza la misma.	

Modificar la evaluación de los trabajos de tesis.

Tabla 4.4. 21 Tarea #1 Historia de usuario Modificar la evaluación de los trabajos de tesis.

Tarea	
Número tarea: 1	Número historia: 16
Nombre tarea: Creación de la interfaz	
Tipo de tarea : Desarrollo	Puntos estimados: 0.33
Fecha inicio: 2 de mayo 2008	Fecha fin: 3 de mayo 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
Descripción: Se crea la interfaz para modificar la evaluación de los trabajos de tesis.	

Tabla 4.4. 22 Tarea #2 Historia de usuario Modificar la evaluación de los trabajos de tesis.

Tarea	
Número tarea: 2	Número historia: 16
Nombre tarea: Selección la evaluación a modificar	
Tipo de tarea : Desarrollo	Puntos estimados: 0.33
Fecha inicio: 3 de mayo 2008	Fecha fin: 4 de mayo 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
Descripción: Se selecciona según el corte que sea de un listado de cortes que dentro tendrán un listado de las tesis dentro de estas están las evaluaciones.	

Tabla 4.4. 23 Tarea #2 Historia de usuario Modificar la evaluación de los trabajos de tesis.

Tarea	
Número tarea: 3	Número historia: 16
Nombre tarea: Actualización de la base de datos	
Tipo de tarea : Desarrollo	Puntos estimados: 0.33
Fecha inicio: 4 de mayo 2008	Fecha fin: 5 de mayo 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
Descripción: Después de haber modificado los campos necesarios se actualiza a la base de datos.	

Eliminar evaluación.

Tabla 4.4. 24 Tarea #1 Historia de usuario Eliminar evaluación.

Tarea	
Número tarea: 1	Número historia: 17
Nombre tarea: Creación de la interfaz	
Tipo de tarea : Desarrollo	Puntos estimados: 0.33
Fecha inicio: 5 de mayo 2008	Fecha fin: 5 de mayo 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	

<p>Descripción: Se crea la interfaz para eliminar la evaluación de los trabajos de tesis.</p>
--

Tabla 4.4. 25 Tarea #2 Historia de usuario Eliminar evaluación.

Tarea	
Número tarea: 2	Número historia: 17
Nombre tarea: Selección la evaluación a eliminar	
Tipo de tarea : Desarrollo	Puntos estimados: 0.33
Fecha inicio: 5 de mayo 2008	Fecha fin: 6 de mayo 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
<p>Descripción: Se selecciona según el corte que sea de un listado de cortes que dentro tendrán un listado de las tesis dentro de estas están las evaluaciones.</p>	

Tabla 4.4. 26 Tarea #3 Historia de usuario Eliminar evaluación.

Tarea	
Número tarea: 3	Número historia: 17
Nombre tarea: Actualización de la base de datos	
Tipo de tarea : Desarrollo	Puntos estimados: 0.33
Fecha inicio: 6 de mayo 2008	Fecha fin: 7 de mayo 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
<p>Descripción: Después de haber eliminado la evaluación se actualiza a la base de datos.</p>	

4.4 Iteración 4

En el transcurso de esta iteración se implementaron las historias de usuario restantes que involucraban las funcionalidades concernientes a los reportes que brinda la aplicación. Al culminar esta, se consta de un producto listo para su puesta en funcionamiento.

Tabla 4.5.1 Historias de usuarios implementadas en la cuarta iteración.

Historias de Usuario	Estimación	Real
Listar profesores	0.4	0.4
Adicionar profesores	0.6	0.6
Eliminar profesores	0.6	0.6
Listar grupos de quinto año	0.4	0.4
Adicionar estudiantes	0.6	0.6
Eliminar estudiantes	0.6	0.6
Listar estudiantes sin tesis	0.4	0.4

Listar profesores

Tabla 4.5.2 Tarea #1 Historia de usuario Listar profesores.

Tarea	
Número tarea: 1	Número historia: 18
Nombre tarea: Creación de la interfaz	
Tipo de tarea : Desarrollo	Puntos estimados: 0.33
Fecha inicio: 7 de mayo 2008	Fecha fin: 8 de mayo 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
Descripción: Se crea la interfaz para listar profesores.	

Tabla 4.5.3 Tarea #2 Historia de usuario Listar profesores.

Tarea	
Número tarea: 2	Número historia: 18
Nombre tarea: Actualizar de la base de datos de la UCI	
Tipo de tarea : Desarrollo	Puntos estimados: 0.33

Fecha inicio: 8 de mayo 2008	Fecha fin: 9 de mayo 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
Descripción: Se actualiza la aplicación de la base de datos de la Universidad.	

Tabla 4.5. 4 Tarea #3 Historia de usuario Listar profesores.

Tarea	
Número tarea: 3	Número historia: 18
Nombre tarea: Mostrar listado de profesores con sus datos.	
Tipo de tarea : Desarrollo	Puntos estimados: 0.33
Fecha inicio: 9 de mayo 2008	Fecha fin: 10 de mayo 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
Descripción: Se muestra un listado de profesores de la facultad con su respectiva carga*(roles que posee en el proceso de tesis) y otros datos.	

Adicionar profesores

Tabla 4.5. 5 Tarea #1 Historia de usuario Adicionar profesores.

Tarea	
Número tarea: 1	Número historia: 19
Nombre tarea: Creación de la interfaz	
Tipo de tarea : Desarrollo	Puntos estimados: 0.33
Fecha inicio: 10 de mayo 2008	Fecha fin: 11 de mayo 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
Descripción: Se crea la interfaz para adicionar profesores.	

Tabla 4.5. 6 Tarea #2 Historia de usuario Adicionar profesores.

Tarea

Número tarea: 2	Número historia: 19
Nombre tarea: Actualización de la base de datos	
Tipo de tarea : Desarrollo	Puntos estimados: 0.33
Fecha inicio: 11 de mayo 2008	Fecha fin: 12 de mayo 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
Descripción: Después de haber adicionado los datos del profesor se actualiza a la base de datos de la aplicación.	

Eliminar profesores.

Tabla 4.5.7 Tarea #1 Historia de usuario Eliminar profesores.

Tarea	
Número tarea: 1	Número historia: 20
Nombre tarea: Creación de la interfaz	
Tipo de tarea : Desarrollo	Puntos estimados: 0.33
Fecha inicio: 12 de mayo 2008	Fecha fin: 13 de mayo 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
Descripción: Se crea la interfaz para eliminar un profesor determinado.	

Tabla 4.5.8 Tarea #2 Historia de usuario Eliminar profesores.

Tarea	
Número tarea: 2	Número historia: 20
Nombre tarea: Actualización de la base de datos	
Tipo de tarea : Desarrollo	Puntos estimados: 0.33
Fecha inicio: 13 de mayo 2008	Fecha fin: 14 de mayo 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
Descripción: Después de haber seleccionado el profesor y seleccionar eliminar se actualiza a la base de datos de la aplicación.	

--

Listar grupos de quinto año.

Tabla 4.5.9 Tarea #1 Historia de usuario Listar grupos de quinto año.

Tarea	
Número tarea: 1	Número historia: 21
Nombre tarea: Creación de la interfaz	
Tipo de tarea : Desarrollo	Puntos estimados: 0.33
Fecha inicio: 14 de mayo 2008	Fecha fin: 15 de mayo 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
Descripción: Se crea la interfaz para listar los grupos de quinto año con sus integrantes.	

Tabla 4.5.10 Tarea #2 Historia de usuario Listar grupos de quinto año.

Tarea	
Número tarea: 2	Número historia: 21
Nombre tarea: Actualizar de la base de datos de la UCI	
Tipo de tarea : Desarrollo	Puntos estimados: 0.33
Fecha inicio: 15 de mayo 2008	Fecha fin: 16 de mayo 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
Descripción: Se actualiza la aplicación de la base de datos de la Universidad.	

Tabla 4.5.11 Tarea #3 Historia de usuario Listar grupos de quinto año.

Tarea	
Número tarea: 3	Número historia: 21
Nombre tarea: Mostrar listado de grupos de quinto año con sus integrantes.	

Tipo de tarea : Desarrollo	Puntos estimados: 0.33
Fecha inicio: 16 de mayo 2008	Fecha fin: 17 de mayo 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
Descripción: Se muestra un listado de grupos de quinto año de la facultad cada uno con un listado de sus respectivos integrantes.	

Adicionar estudiantes.

Tabla 4.5. 12 Tarea #1 Historia de usuario Adicionar estudiantes.

Tarea	
Número tarea: 1	Número historia: 22
Nombre tarea: Creación de la interfaz	
Tipo de tarea : Desarrollo	Puntos estimados: 0.33
Fecha inicio: 18 de mayo 2008	Fecha fin: 18 de mayo 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
Descripción: Se crea la interfaz para adicionar un estudiante.	

Tabla 4.5. 13 Tarea #2 Historia de usuario Adicionar estudiantes.

Tarea	
Número tarea: 2	Número historia: 22
Nombre tarea: Adición de los datos del estudiante a la base de datos	
Tipo de tarea : Desarrollo	Puntos estimados: 0.33
Fecha inicio: 18 de mayo 2008	Fecha fin: 19 de mayo 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
Descripción: Después de haber llenado los datos del estudiante se agrega a la base de datos y se actualiza la misma.	

Eliminar estudiantes.

Tabla 4.5. 14 Tarea #1 Historia de usuario Eliminar estudiantes.

Tarea	
Número tarea: 1	Número historia: 23
Nombre tarea: Creación de la interfaz	
Tipo de tarea : Desarrollo	Puntos estimados: 0.33
Fecha inicio: 19 de mayo 2008	Fecha fin: 20 de mayo 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
Descripción: Se crea la interfaz para eliminar un estudiante.	

Tabla 4.5. 15 Tarea #2 Historia de usuario Eliminar estudiantes.

Tarea	
Número tarea: 2	Número historia: 23
Nombre tarea: Eliminar el estudiante de la base de datos.	
Tipo de tarea : Desarrollo	Puntos estimados: 0.33
Fecha inicio: 20 de mayo 2008	Fecha fin: 21 de mayo 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
Descripción: Después de haber seleccionado el estudiante se elimina de la base de datos y se actualiza la misma.	

Listar estudiantes sin tesis.

Tabla 4.5. 16 Tarea #1 Historia de usuario listar estudiantes sin tesis.

Tarea	
Número tarea: 1	Número historia: 24

Nombre tarea: Creación de la interfaz	
Tipo de tarea : Desarrollo	Puntos estimados: 0.33
Fecha inicio: 22 de mayo 2008	Fecha fin: 23 de mayo 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
Descripción: Se crea la interfaz para listar estudiantes sin tesis.	

Tabla 4.5. 17 Tarea #2 Historia de usuario listar estudiantes sin tesis.

Tarea	
Número tarea: 2	Número historia: 24
Nombre tarea: Chequear la consulta a la base de datos.	
Tipo de tarea : Desarrollo	Puntos estimados: 0.33
Fecha inicio: 23 de mayo 2008	Fecha fin: 24 de mayo 2008
Programador responsable: Emilio Salterio Paz – Reynaldo Heredia Rodríguez	
Descripción: Se hace una consulta a la base de datos y se muestra el listado de estudiantes sin tesis.	

4.6 Pruebas

Las características del software que no pueden ser demostradas mediante pruebas simplemente no existen. Las pruebas dan la oportunidad de saber si lo implementado es lo que en realidad se tenía en mente. Las pruebas nos indican que nuestro trabajo funciona, cuando no podemos pensar en ninguna prueba que pudiese originar un fallo en nuestro sistema, entonces habremos acabado por completo.

Uno de los pilares fundamentales de XP es el proceso de pruebas (Beck, 2000). Mediante este método se reduce el número de errores no detectados así como el tiempo entre la introducción de este en el sistema y su detección (Crispin y House, 2002). Todo esto contribuye a elevar la calidad de los

productos desarrollados y la seguridad de los programadores a la hora de introducir cambios y modificaciones.

XP divide las pruebas en dos grupos: pruebas unitarias, desarrolladas por los programadores y encargadas de verificar el código de forma automática y las pruebas de aceptación, destinadas a evaluar si al final de una iteración se consiguió la funcionalidad requerida además de comprobar que dicha funcionalidad sea la esperada por el cliente (Crispin y House, 2002).

4.7 Pruebas de Aceptación

Las pruebas de aceptación son pruebas de caja negra que se crean a partir de las historias de usuario (Crispin y House, 2002). Durante las iteraciones las historias de usuario seleccionadas serán traducidas a pruebas de aceptación. En ellas se especifican, desde la perspectiva del cliente, los escenarios para probar que una historia de usuario ha sido implementada correctamente. Una historia de usuario puede tener todas las pruebas de aceptación que necesite para asegurar su correcto funcionamiento. El objetivo final de estas es garantizar que los requerimientos han sido cumplidos y que el sistema es aceptable (XP, 2006). Las pruebas de aceptación realizadas al sistema pueden verse en el anexo II.

4.8 Diagramas de clases

Para el diseño de las aplicaciones, la metodología XP no requiere la representación del sistema mediante diagramas de clases utilizando notación UML, en su lugar se usan otras técnicas como las tarjetas CRC (XP, 2006). No obstante el uso de estos puede aplicarse siempre y cuando mejore la comunicación, no sea un peso su mantenimiento, no sean extensos y se enfoquen en la información importante (Fowler, 2004). Aunque el sistema no fue desarrollado orientado a objeto por lo que no se utilizaron clases en su desarrollo, ya que Drupal no es orientado a objetos, si se generaron diagramas de la base de datos que se muestran en el anexo I.

Conclusiones

En el presente capítulo se hace alusión a las etapas de implementación y pruebas del software en desarrollo. Para ello se exponen todos los artefactos generados, realizando una descripción de cada uno de ellos.

Capítulo V. Estudio de Factibilidad

5.1 Introducción.

Una de las tareas de mayor importancia en la planificación de proyectos de software es la estimación, la cual consiste en determinar con cierto grado de certeza, los recursos de hardware y software, costo, tiempo y esfuerzo necesarios para el desarrollo de los mismos. En el presente capítulo se realiza un estudio de factibilidad para la realización del sistema propuesto, haciendo una estimación del esfuerzo necesario para llevar a cabo el mismo.

En este capítulo se utilizará COCOMO II (Constructive Cost Model) para realizar el estudio de factibilidad del producto analizado, el cual se basa en el uso de ecuaciones matemáticas que permiten calcular el esfuerzo a partir de ciertas métricas de tamaño estimado, como el análisis de puntos de función y las líneas de código fuente (en inglés SLOC, Source Line Of Code).

5.2 Características del proyecto.

Tabla 5.1.1 Entradas Externas.

Nombre de la entrada externa	Cantidad de ficheros	Cantidad de elementos de datos	Clasificación (simple, media y compleja)
Autenticar usuario.	1	2	Simple
Crear Comité de tesis.	1	1	Simple
Modificar Comité de tesis.	1	1	Simple
Crear Tribunal	1	6	Simple
Modificar tribunal.	1	5	Simple
Proponer tesis.	1	3	Simple
Modificar tema de tesis.	1	3	Simple
Editar evaluación.	1	4	Simple
Adicionar profesor.	1	2	Simple
Adicionar estudiante.	1	1	Simple
Total		28	

Tabla 5.1. 2 Salidas externas.

Nombre de las salidas externa	Cantidad de ficheros	Cantidad de elementos de datos	Clasificación (simple, media y compleja)
Lista de profesores	12	1	Simple
Sin tesis	10	1	Simple
Cronograma	1	1	Simple
Grupos	1	1	Simple
Total		4	

Tabla 5.1. 3 Ficheros internos.

Nombre del fichero interno	Cantidad de ficheros	Cantidad de elementos de datos	Clasificación (simple, media y compleja)
Aprobar Tema de tesis.	1	5	Simple
Asignar Tema de Tesis.	1	8	Simple
Evaluar Trabajo de Tesis.	1	4	Simple
Eliminar Comité de tesis.	1	7	Simple
Eliminar tribunal.	1	1	Simple
Eliminar usuario.	2	2	Simple
Eliminar Tema de tesis.	1	1	Simple
Eliminar evaluaciones.	1	1	Simple
Eliminar profesores.	1	1	Simple
Eliminar estudiante.	1	1	Simple
Total		31	

Tabla 5.1. 4 Peticiones.

Nombre de la petición	Cantidad de ficheros	Cantidad de elementos de datos	Clasificación (simple, media y compleja)
Total		0	

Tabla 5.1. 5 Interfaces externas.

Nombre de la Interfaz externa	Cantidad de ficheros	Cantidad de elementos de datos	Clasificación (simple, media y compleja)
Total		0	

Tabla 5.1. 6 Puntos de función desajustados.

Elementos	Simple		Medio		Complejo		Subtotal
	No.	Peso	No.	Peso	No.	Peso	
Entradas Externas.	28	3	0	4	0	6	84
Salidas externas.	4	4	0	5	0	7	16
Ficheros internos.	31	7	0	10	0	15	217
Peticiones.	0	3	0	4	0	6	0
Interfaces externas.	0	5	0	7	0	10	0
Total							317

5.3 Cálculo de instrucciones fuentes, esfuerzo, tiempo de desarrollo, cantidad de hombres y costo.

El modelo COCOMO II usa Puntos de Función y/o Líneas de Código Fuente (SLOC) como base para medir tamaño en los modelos de estimación de Diseño Temprano y Post-Arquitectura. Los Puntos de Función procuran cuantificar la funcionalidad de un sistema de software. La meta es obtener un número que caracterice completamente al sistema. Son útiles estimadores ya que están basados en información que está disponible en las etapas tempranas del ciclo de vida del desarrollo de software. COCOMO II considera solamente UFP (del inglés: Puntos Función Desajustados).

Tabla 5.2. 1 Características.

Características	Valor
Puntos de función desajustados	317

Lenguaje (PHP)	30
Instrucciones fuentes por puntos de función	16801
Instrucciones fuentes	9.51 ksloc

Tabla 5.2. 2 Factores de escala.

Nombre	Valor	Justificación
PREC	1.24	Existen proyectos similares a nivel de universidad.
FLEX	2.03	Cuenta con una alta flexibilidad.
RESL	4.24	No se identificaron riesgos críticos.
TEAM	2.19	El equipo presenta una alta cohesión.
PMAT	4.68	Existe poca experiencia previa en el desarrollo de estos productos
Total(SF)	14.38	

Tabla 5.2. 3 Multiplicadores de esfuerzo

Nombre	Valor	Justificación
RCPX	1	Las exigencias de confiabilidad, documentación y volumen de datos son moderadas, y la complejidad del producto es media.
RUSE	0.95	No se pretende reutilizar nada.
PDIF	0.87	No existen restricciones en cuando al tiempo de CPU o al consumo de memoria, la plataforma es muy estable.
PREX	1.33	Poca experiencia en el desarrollo sobre la plataforma y el lenguaje.
PERS	0.88	Alta capacidad del personal.
FCIL	0.82	Se utilizan entornos de desarrollo integrados y herramientas de modelación que facilitan el trabajo.
FCED	1.00	Se empleó el tiempo planificado para el desarrollo de la aplicación.
Total(EM)	0.98	

Cálculos

$$A = 2.94 \quad B = 1.05 \quad C = 3.67 \quad D = 0.24 \quad SF = 14.38 \quad EM = 0.98$$

$$B = 0.91 + 0.01 * SF = 0.91 + 0.01 * 14.38 = 1.05$$

$$E = B + 0.01 * SF = 1.05 + 0.01 * 14.38 = 1.19$$

$$PM = A * Size^E * \Gamma EM = 2.94 * 9.51^{1.19} * 0.98 = 42.04$$

$$F = D + 0.2 * 0.01 * \Sigma SF = 0.24 + 0.2 * 0.01 * 14.38 = 0.27$$

$$TDEV = C * PM^F = 3.67 * 42.04^{0.27} = 10.05$$

$$CH = PM / TDEV = 42.04 / 10.05 = 4$$

$$C = CH * Sal * PM = 4 * 100 * 42.04 = 16816$$

Tabla 5.2. 4 Resultados.

Cálculo de:	Valor
Esfuerzo	42.04Hombres/mes
Tiempo de desarrollo	10.1 meses
Cantidad de hombres	2 hombres
Salario medio	\$ 100
Costo	\$16816

Beneficios tangibles e intangibles.

El desarrollo del producto propuesto en este trabajo brinda como beneficio principal a la Facultad 8 Soluciones a las limitantes presentadas por la no existencia en la facultad de una aplicación Web dinámica que ayude a eliminar el trabajo manual referente al proceso de trabajo de tesis, permitiendo economizar gran parte del tiempo que se dedicaba a impresión de reportes de información a profesores y estudiantes o a la digitalización de la infamación generada en cortes evaluativos u otras actividades referentes a este proceso. Esta aplicación no está concebida para ser un producto

comercial, aunque en un futuro pueda reportar beneficios monetarios a la Universidad adquiriendo este carácter, por la posible venta de la misma a otras universidades para su posterior uso.

Análisis de costo.

El desarrollo de un producto siempre tiene un costo de producción, el cual debe ser justificado en base a los beneficios reportados por el mismo. El sistema que se propone en este trabajo no conlleva a grandes gastos, puesto que solo es influyente el salario de los desarrolladores, por lo cual se concluye que su implementación es factible. Esto se debe en gran medida a la utilización de plataformas, APIs y herramientas libres que no requieren el pago de alguna licencia.

Conclusiones

En el presente capítulo se realizó un análisis de factibilidad de la solución propuesta, arribando a la conclusión de que es viable su desarrollo comparando los costos de producción con los beneficios reportados por su puesta en funcionamiento.

Conclusiones generales

- La implementación del sistema resuelve las limitantes presentadas anteriormente y pone en manos de la facultad 8 una herramienta que permite automatizar el proceso relacionado con la automatización de toda la información generada en el proceso de tesis de grado.
- En la actualidad es muy bajo el nivel de utilización de sistemas de gestión de información de este tipo en Cuba. En los centros estudiados el proceso se realiza de forma manual retardando así el proceso y generando más trabajo y gasto de recursos.
- El sistema desarrollado es independiente de la forma en que se ejecute el proceso en las distintas universidades, así como del sistema operativo del servidor en que se ejecute.
- Se seleccionaron las herramientas necesarias para realizar el proyecto con la calidad requerida y que estuvieran a tono con las disposiciones existentes, siempre que fuera posible, sobre el uso de software libre y multiplataforma.
- El sistema propuesto cuenta con un mecanismo de notificación mediante el envío de correos electrónicos a los profesores que se encuentran registrados en la aplicación, permitiendo su fácil ampliación para brindar soporte a otros canales de comunicación.
- El sistema cuenta con una interfaz *Web* amigable que muestra información referente a las tesis realizadas, además de permitir la administración del mismo.

Recomendaciones

- Adicionar más funcionalidades para la mejora del trabajo y para brindar más información del proceso en cuestión.
- Desarrollar *plugins* para incrementar las capacidades del sistema.
- En versiones futuras poder generar reportes en formato Excel.
- En versiones futuras crear un repositorio de bases de datos para poder consultar informaciones de años anteriores.
- Compartir esta idea para el uso de las demás facultades e intentar comercializarlo con otras universidades del país ayudando así con la digitalización del país.

Bibliografía

- What is content management system? - a definition from Whatis.com - see also: content managemen. *SearchSOA.com* September 2007. [cited 22 May 2008]. Available from world wide web: <http://searchsoa.techtarget.com/sDefinition/0,,sid26_gc i508916,00.html>.
- Aplcaciones Web. » ventaja e inconvenientes de las aplicaciones web. *Ventajas e inconvenientes de las aplicaciones web* September 2007. [cited 23 May 2008]. Available from world wide web: <<http://www.avidos.net/blogold/aplicaciones-web/>>.
- Beck, K, and Wesley Addison . *Extreme Programming Explained.*. 2000, September 2007.
- Computación Aplicada al Desarrollo SA de CV. Historia del lenguaje Java. September 2007. [cited 22 May 2008]. Available from world wide web: <http://www.cad.com.mx/historia_del_lenguaje_java.htm>.
- E.Rivera. Asesoría informática - Diccionario definiciones informáticas. *Asesoría informática* September 2007. [cited 22 May 2008]. Available from world wide web: <http://www.asesoriainformatica.com/definiciones_c.htm>.
- Fernández Escribano, Gerardo. Presentacion-XP.pdf (application/pdf Object). *Introducción a Extreme Programming* September 2007. [cited 23 May 2008]. Available from world wide web: <<http://www.info-ab.uclm.es/asignaturas/42551/trabajosAnteriores/Presentacion-XP.pdf>>.
- Graphicsperu. Glosarios de Terminos abcde en internet en Peru - GraphicsPeru. *Glosario de terminos* September 2007. [cited 22 May 2008]. Available from world wide web: <http://www.graphicsperu.com/glosario_grafico.htm>.
- HTML.it s.r.l. - Design. HTMLpoint - Tutorial ISS - Internet Information Server. *HTMLPOINT.com* September 2007. [cited 23 May 2008]. Available from world wide web: <<http://www.htmlpoint.com/iis/index.html>>.

- IBPhoenix. Firebird-1.5-Arranque.pdf (application/pdf Object). *Firebird 1.5 - Guía de arranque rápido* September 2007. [cited 23 May 2008]. Available from world wide web: <<http://www.firebirdsql.org/pdfmanual/es/Firebird-1.5-Arranque.pdf>>.
- Kirily "Skud" Robert <skud@cpan.org>. perlintro - perldoc.perl.org. *perldoc.perl.org* September 2007. [cited 22 May 2008]. Available from world wide web: <<http://perldoc.perl.org/perlintro.html>>.
- lighttpd fly light. lighttpd fly light. *Lighttpd server* September 2007. [cited 23 May 2008]. Available from world wide web: <<http://www.lighttpd.net/>>.
- Metodologías De Desarrollo De Software. Capítulo 2. IAGP 2005/06. Metodologías de desarrollo de software. *Capítulo 2. IAGP 2005/06. Metodologías de desarrollo de software* September 2007. [cited 23 May 2008]. Available from world wide web: <<http://www.um.es/docencia/barzana/IAGP/Iagp2.html>>.
- MySQL. MySQL Internals - MySQL Forge Wiki. *Resources for the MySQL Community* September 2007. [cited 23 May 2008]. Available from world wide web: <http://forge.mysql.com/wiki/MySQL_Internals>.
- MySQL AB. MySQL :: MySQL 5.0 Reference Manual. *The world's most popular open source database* September 2007. [cited 23 May 2008]. Available from world wide web: <<http://dev.mysql.com/doc/refman/5.0/es/index.html>>.
- PHP en la web. El lenguaje PHP y algunas aplicaciones prácticas. *El lenguaje PHP y algunas aplicaciones prácticas* September 2007. [cited 23 May 2008]. Available from world wide web: <http://www.efaber.net/formacion/fp/curso_php/index.html>.
- PostgreSQL. Tutorial de PostgreSQL. *Tutorial de PostgreSQL* September 2007. [cited 23 May 2008]. Available from world wide web: <<http://es.tldp.org/Postgresql-es/web/navegable/tutorial/tutorial.html>>.
- Sitio Dupal. Sobre Drupal | Drupal Hispano. *Comunidad de usuarios de Drupal* September 2007. [cited 22 May 2008]. Available from world wide web: <<http://www.drupal.org/es/drupal>>.

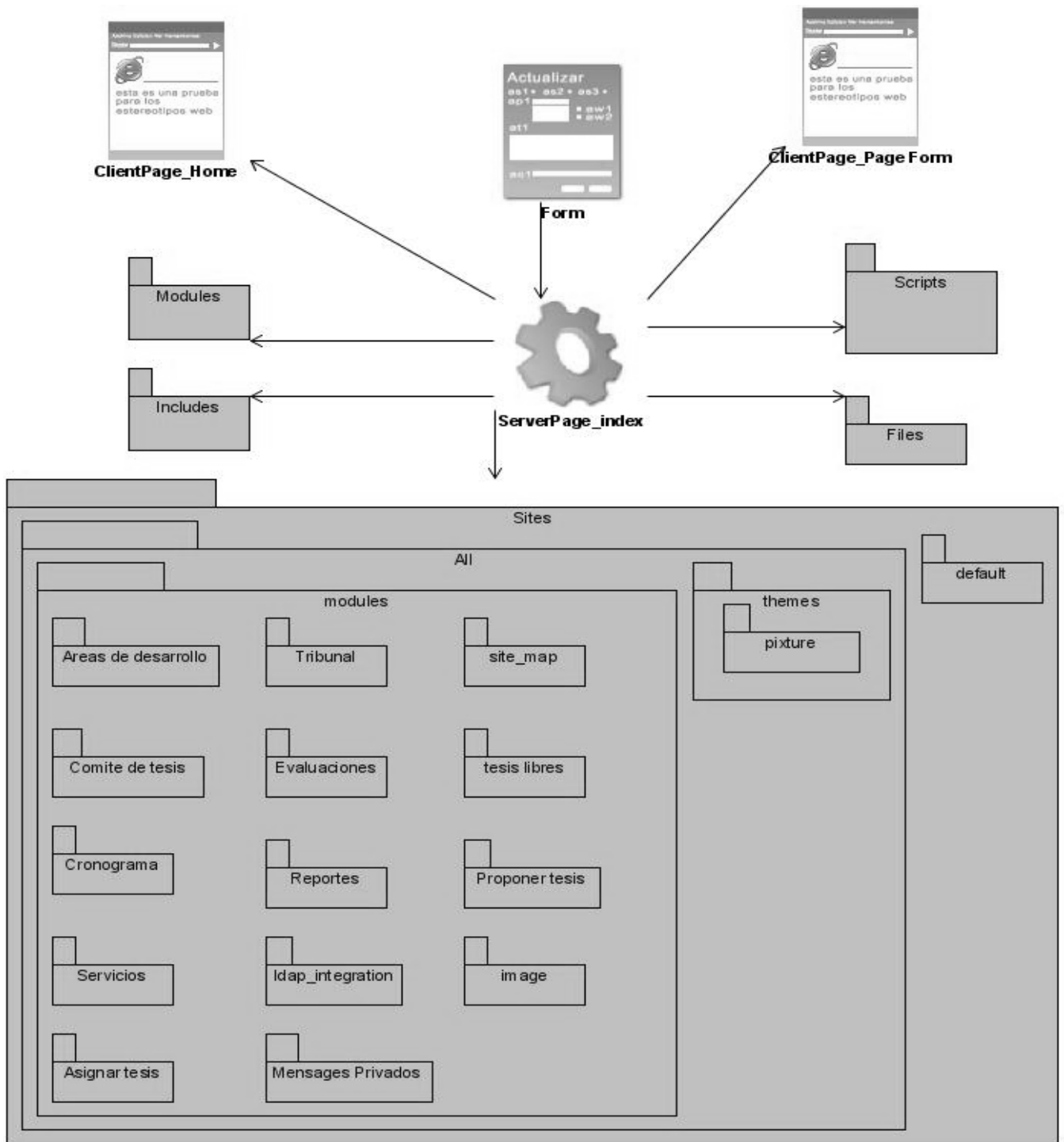
The Apache Software Foundation. Documentación del Servidor HTTP Apache 2.0 - Servidor HTTP Apache. *Documentación del Servidor HTTP Apache 2.0* September 2007. [cited 23 May 2008]. Available from world wide web: <<http://httpd.apache.org/docs/2.0/es/>>.

THTTPD. thttpd. *thttpd - tiny/turbo/throttling HTTP server* September 2007. [cited 23 May 2008]. Available from world wide web: <<http://acme.com/software/thttpd/>>.

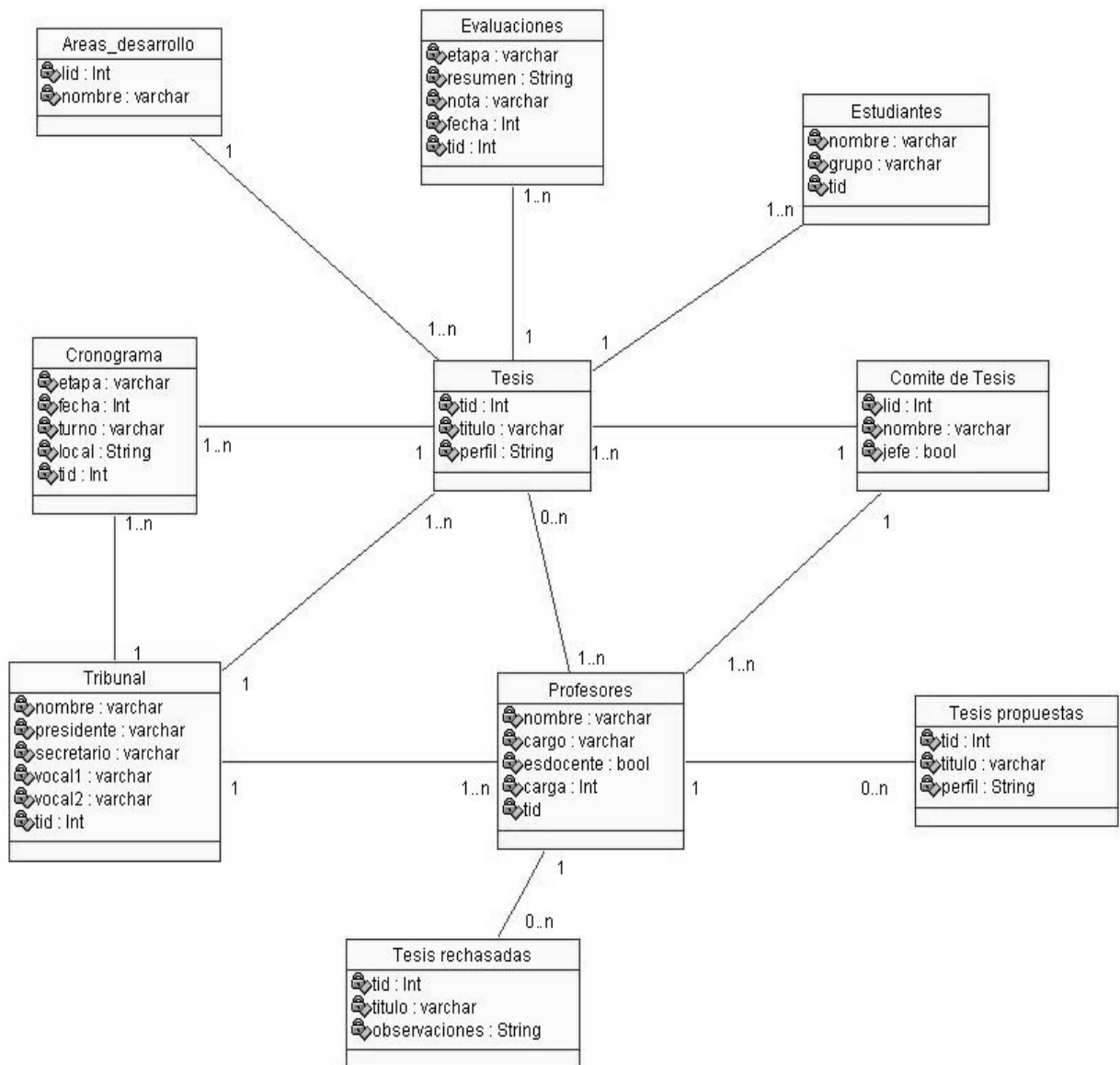
Wikipedia® . Sistema de gestión de contenido - Wikipedia, la enciclopedia libre. *Sistema de gestión de contenido, Distintos tipos de CMS* September 2007. [cited 22 May 2008]. Available from world wide web: <http://es.wikipedia.org/wiki/Sistema_de_gesti%C3%B3n_de_contenido>.

Xavier Cuerda García, and Julià Minguillón Alfonso . Introducción a los Sistemas de Gestión de Contenidos (CMS) de código abierto. *Introducción a los Sistemas de Gestión de Contenidos (CMS) de código abierto* September 2007. [cited 22 May 2008]. Available from world wide web: <<http://mosaic.uoc.edu/articulos/cms1204.html>>.

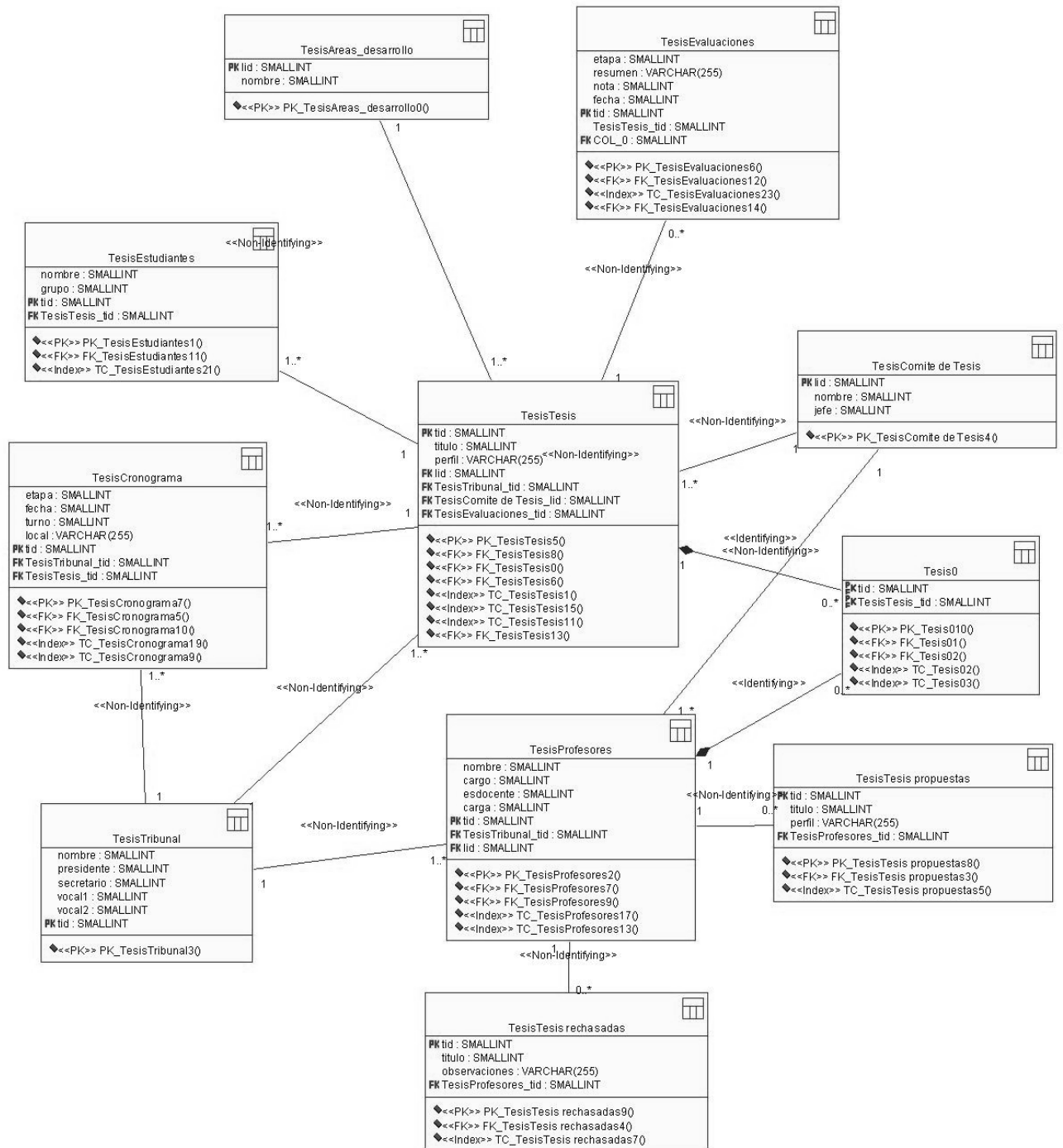
Anexo I Diagramas de clases.



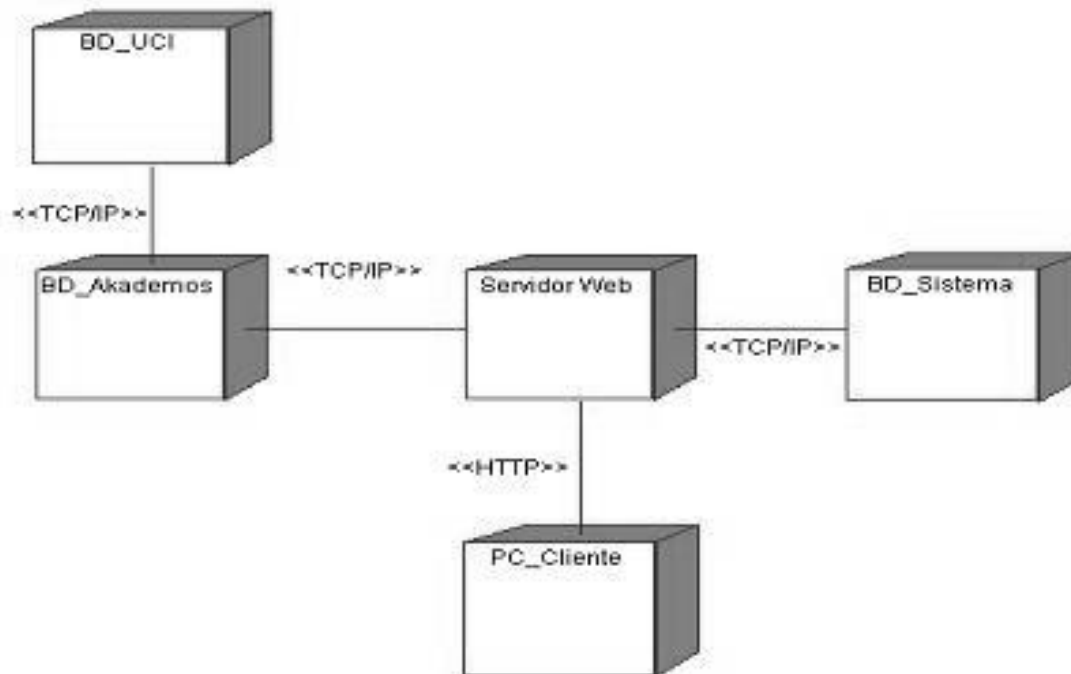
Anexo I Figura 4.6.1 Diagrama de clases de estereotipos Web



Anexo I Figura 4.6. 2 Diagrama de Clases persistentes



Anexo I Figura 4.6.3 Diagrama de la base de datos.



Anexo I Figura 4.6. 4 Diagrama de Despliegue del sistema

Anexo II Pruebas de Aceptación.

Anexo II Tabla 4.7. 1 Pruebas de aceptación para la historia de usuario Autenticar usuario.

Caso de Prueba de Aceptación	
Código: HU1_P1	Historia de Usuario: Autenticar Usuario
Nombre: Autenticar usuario en el sistema.	
Descripción: Probar que se autentique correctamente el usuario en el sistema	
Condiciones de Ejecución: La aplicación debe ser ejecutada normalmente, la contraseña y el usuario deben de verificarse en la base de datos de la UC1.	
Entrada / Pasos de ejecución: Se intenta autenticar un usuario con datos válidos.	
Resultado Esperado: El usuario es reconocido y logueado correctamente en el sistema	
Evaluación de la Prueba: Prueba satisfactoria.	

Anexo II Tabla 4.7. 2 Pruebas de aceptación para la historia de usuario Autenticar usuario.

Caso de Prueba de Aceptación	
Código: HU1_P2	Historia de Usuario: Autenticar Usuario
Nombre: Autenticar usuario en el sistema.	
Descripción: Probar que no se autentique un usuario en el sistema con datos no validos.	
Condiciones de Ejecución: La aplicación debe ser ejecutada normalmente, la contraseña y el usuario deben de verificarse en la base de datos de la UC1.	
Entrada / Pasos de ejecución: Se intenta autenticar un usuario con datos no válidos.	
Resultado Esperado: El usuario no es reconocido y no puede acceder correctamente en el sistema.	
Evaluación de la Prueba: Prueba satisfactoria.	

Anexo II Tabla 4.7. 3 Pruebas de aceptación para la historia de usuario Eliminar usuario.

Caso de Prueba de Aceptación	
Código: HU2_P1	Historia de Usuario: Eliminar Usuario
Nombre: Eliminar privilegios de un usuario en el sistema.	
Descripción: Probar que se elimine correctamente los privilegios de un usuario en el sistema	
Condiciones de Ejecución: La aplicación debe ser ejecutada con privilegios de administración, los privilegios de dicho usuario deben de borrarse de la base de datos del sistema.	
Entrada / Pasos de ejecución: Se intenta eliminar los privilegios de un usuario con datos válidos dentro de la aplicación.	
Resultado Esperado: Los privilegios del usuario son eliminados correctamente de la base de datos del sistema	
Evaluación de la Prueba: Prueba satisfactoria.	

Anexo II Tabla 4.7. 4 Pruebas de aceptación para la historia de usuario Crear Comité de tesis.

Caso de Prueba de Aceptación	
Código: HU3_P1	Historia de Usuario: Crear Comité de tesis.
Nombre: Insertar los datos de un comité de tesis en el sistema.	
Descripción: Probar que se adicionen correctamente los datos de un comité de tesis en la base de datos de la aplicación.	
Condiciones de Ejecución: La aplicación debe ser ejecutada con privilegios de administración, los datos de dicho comité de tesis deben de guardarse en la base de datos del sistema.	
Entrada / Pasos de ejecución: Se intenta adicionar los datos de un comité de tesis en la aplicación la aplicación.	
Resultado Esperado: Los datos del comité de tesis son adicionados correctamente en la base de datos del sistema	
Evaluación de la Prueba: Prueba satisfactoria.	

Anexo II Tabla 4.7. 5 Pruebas de aceptación para la historia de usuario Modificar Comité de tesis.

Caso de Prueba de Aceptación	
Código: HU4_P1	Historia de Usuario: Modificar Comité de tesis.
Nombre: Modificar los datos de un comité de tesis en el sistema.	
Descripción: Probar que se modifiquen correctamente los datos de un comité de tesis en la base de datos de la aplicación.	
Condiciones de Ejecución: La aplicación es ejecutada con privilegios de administración, los datos de dicho comité de tesis deben de poder modificarse en la base de datos del sistema.	
Entrada / Pasos de ejecución: Se intenta modificar los datos de un comité de tesis en la aplicación la aplicación.	
Resultado Esperado: Los datos del comité de tesis son modificados en la base de datos del sistema	
Evaluación de la Prueba: Prueba satisfactoria.	

Anexo II Tabla 4.7. 6 Pruebas de aceptación para la historia de usuario Eliminar Comité de tesis.

Caso de Prueba de Aceptación	
Código: HU5_P1	Historia de Usuario: Eliminar Comité de tesis.
Nombre: Eliminar los datos de un comité de tesis en el sistema.	
Descripción: Probar que se Eliminen correctamente los datos de un comité de tesis en la base de datos de la aplicación.	
Condiciones de Ejecución: La aplicación es ejecutada con privilegios de administración, los datos de dicho comité de tesis deben de poder ser eliminados en la base de datos del sistema.	
Entrada / Pasos de ejecución: Se intenta eliminar los datos del comité de tesis en la aplicación la aplicación.	
Resultado Esperado: Los datos del comité de tesis son eliminados en la base de datos del sistema	

Evaluación de la Prueba: Prueba satisfactoria.

Anexo II Tabla 4.7. 7 Pruebas de aceptación para la historia de usuario Crear Tribunal.

Caso de Prueba de Aceptación	
Código: HU6_P1	Historia de Usuario: Crear tribunal.
Nombre: Adicionar los datos de un tribunal en el sistema.	
Descripción: Probar que se adicionen correctamente los datos de un tribunal en la base de datos de la aplicación.	
Condiciones de Ejecución: La aplicación es ejecutada con privilegios de administración, los datos de dicho tribunal deben de poder ser adicionados en la base de datos del sistema.	
Entrada / Pasos de ejecución: Se intenta adicionar los datos del tribunal en la aplicación la aplicación.	
Resultado Esperado: Los datos del tribunal son adicionados correctamente en la base de datos del sistema	
Evaluación de la Prueba: Prueba satisfactoria.	

Anexo II Tabla 4.7. 8 Pruebas de aceptación para la historia de usuario Modificar Tribunal.

Caso de Prueba de Aceptación	
Código: HU7_P1	Historia de Usuario: Modificar tribunal.
Nombre: Modificar los datos de un tribunal en el sistema.	
Descripción: Probar que se modifiquen correctamente los datos de un tribunal en la base de datos de la aplicación.	
Condiciones de Ejecución: La aplicación es ejecutada con privilegios de administración, los datos de dicho tribunal deben de poder ser modificados en la base de datos del sistema.	
Entrada / Pasos de ejecución: Se intenta modificar los datos del tribunal en la aplicación la aplicación.	
Resultado Esperado: Los datos del tribunal son modificados correctamente en la base de datos del sistema	

Evaluación de la Prueba: Prueba satisfactoria.

Anexo II Tabla 4.7. 9 Pruebas de aceptación para la historia de usuario Eliminar Tribunal.

Caso de Prueba de Aceptación	
Código: HU8_P1	Historia de Usuario: Eliminar tribunal.
Nombre: Eliminar los datos de un tribunal en el sistema.	
Descripción: Probar que se eliminen correctamente los datos de un tribunal en la base de datos de la aplicación.	
Condiciones de Ejecución: La aplicación es ejecutada con privilegios de administración, los datos de dicho tribunal deben de poder ser eliminados en la base de datos del sistema.	
Entrada / Pasos de ejecución: Se intenta eliminar los datos del tribunal en la aplicación la aplicación.	
Resultado Esperado: Los datos del tribunal son eliminados correctamente en la base de datos del sistema	
Evaluación de la Prueba: Prueba satisfactoria.	

Anexo II Tabla 4.7. 10 Pruebas de aceptación para la historia de usuario Publicar tema de tesis.

Caso de Prueba de Aceptación	
Código: HU9_P1	Historia de Usuario: Publicar tema de tesis
Nombre: Adicionar los datos de un tema de tesis propuesto en el sistema.	
Descripción: Probar que se adicionen correctamente los datos de un tema de tesis propuesto en la base de datos de la aplicación.	
Condiciones de Ejecución: La aplicación es ejecutada por un usuario que sea identificado como profesor, los datos de dicho tema de tesis deben de poder ser adicionados en la base de datos del sistema.	
Entrada / Pasos de ejecución: Se intenta adicionar los datos del tema de tesis propuesto en la aplicación la aplicación.	
Resultado Esperado: Los datos del tema de tesis son adicionados	

correctamente en la base de datos del sistema
Evaluación de la Prueba: Prueba satisfactoria.

Anexo II Tabla 4.7. 11 Pruebas de aceptación para la historia de usuario Eliminar Tema de tesis.

Caso de Prueba de Aceptación	
Código: HU10_P1	Historia de Usuario: Eliminar tema de tesis
Nombre: Eliminar los datos de un tema de tesis (propuesto o rechazado) en el sistema.	
Descripción: Probar que se eliminen correctamente los datos de un tema de tesis (propuesto o rechazado) de la base de datos de la aplicación.	
Condiciones de Ejecución: La aplicación es ejecutada con privilegios de administrador, los datos de dicho tema de tesis deben de poder ser eliminados en la base de datos del sistema.	
Entrada / Pasos de ejecución: Se intenta eliminar los datos del tema de tesis (propuesto o rechazado) en la aplicación la aplicación.	
Resultado Esperado: Los datos del tema de tesis son eliminados correctamente en la base de datos del sistema	
Evaluación de la Prueba: Prueba satisfactoria.	

Anexo II Tabla 4.7. 12 Pruebas de aceptación para la historia de usuario Modificar tema de tesis.

Caso de Prueba de Aceptación	
Código: HU11_P1	Historia de Usuario: Modificar tema de tesis.
Nombre: Modificar los datos de un tema de tesis (propuesto o rechazado) en el sistema.	
Descripción: Probar que se modifiquen correctamente los datos de un tema de tesis (propuesto o rechazado) en la base de datos de la aplicación.	
Condiciones de Ejecución: La aplicación es ejecutada por un usuario que sea identificado como profesor, los datos de dicho tema de tesis deben de poder ser modificados en la base de	

datos del sistema.
Entrada / Pasos de ejecución: Se intenta modificar los datos de un tema de tesis (propuesto o rechazado) en la aplicación la aplicación.
Resultado Esperado: Los datos del tema de tesis son modificados correctamente en la base de datos del sistema
Evaluación de la Prueba: Prueba satisfactoria.

Anexo II Tabla 4.7. 13 Pruebas de aceptación para la historia de usuario Aprobar Tema de tesis.

Caso de Prueba de Aceptación	
Código: HU12_P1	Historia de Usuario: Aprobar tema de tesis.
Nombre: Aprobar un tema de tesis (propuesto o rechazado) en el sistema.	
Descripción: Probar que se adicionen correctamente los datos de un tema de tesis aprobado en la base de datos de la aplicación.	
Condiciones de Ejecución: La aplicación es ejecutada con privilegios de administración, los datos de dicho tema de tesis deben de poder ser adicionados en la base de datos del sistema.	
Entrada / Pasos de ejecución: Se intenta aprobar un tema de tesis (propuesto o rechazado) en la aplicación la aplicación.	
Resultado Esperado: Los datos del tema de tesis son adicionados correctamente en la base de datos del sistema	
Evaluación de la Prueba: Prueba satisfactoria.	

Anexo II Tabla 4.7. 14 Pruebas de aceptación para la historia de usuario Asignar Tema de Tesis.

Caso de Prueba de Aceptación	
Código: HU13_P1	Historia de Usuario: Asignar tema de tesis.
Nombre: Asignar un tema de tesis aprobado.	
Descripción: Probar que se adicionen correctamente los datos de un tema de tesis aprobado que haya sido asignado, en la base de datos de la aplicación.	

<p>Condiciones de Ejecución: La aplicación es ejecutada con privilegios de administración, los datos de dicho tema de tesis deben de poder ser adicionados en la base de datos del sistema.</p>
<p>Entrada / Pasos de ejecución: Se intenta asignar un tema de tesis aprobado a estudiantes de quinto año, en la aplicación la aplicación.</p>
<p>Resultado Esperado: Los datos del tema de tesis asignado son adicionados correctamente en la base de datos del sistema</p>
<p>Evaluación de la Prueba: Prueba satisfactoria.</p>

Anexo II Tabla 4.7. 15 Pruebas de aceptación para la historia de usuario Evaluar Trabajo de Tesis.

Caso de Prueba de Aceptación	
Código: HU14_P1	Historia de Usuario: Evaluar trabajo de tesis.
Nombre: Evaluar trabajo de tesis.	
Descripción: Probar que se adicionen correctamente los datos de una evaluación a un trabajo de tesis, en la base de datos de la aplicación.	
<p>Condiciones de Ejecución: La aplicación es ejecutada con privilegios de administración, los datos de la evaluación de dicho trabajo de tesis deben de poder ser adicionados en la base de datos del sistema.</p>	
Entrada / Pasos de ejecución: Se intenta adicionar una evaluación de un trabajo de tesis, en la aplicación la aplicación.	
Resultado Esperado: Los datos de la evaluación, son adicionados correctamente en la base de datos del sistema	
Evaluación de la Prueba: Prueba satisfactoria.	

Anexo II Tabla 4.7. 16 Pruebas de aceptación para la historia de usuario Listar profesores.

Caso de Prueba de Aceptación	
Código: HU15_P1	Historia de Usuario: Listar profesores.
Nombre: Comprobar el reporte listar profesores	

Descripción: Probar que se muestran correctamente los datos del reporte listar profesores en la aplicación.
Condiciones de Ejecución: La aplicación es ejecutada por cualquier usuario del dominio, los datos del reporte listar profesores deben ser mostrados correctamente por el sistema.
Entrada / Pasos de ejecución: Se intenta mostrar el reporte listar profesores
Resultado Esperado: Los datos del reporte son mostrados correctamente en la aplicación
Evaluación de la Prueba: Prueba satisfactoria.

Anexo II Tabla 4.7. 17 Pruebas de aceptación para la historia de usuario adicionar profesor.

Caso de Prueba de Aceptación	
Código: HU16_P1	Historia de Usuario: Adicionar profesor
Nombre: Comprobar que adiciona profesor	
Descripción: Probar que se adiciona correctamente los datos del profesor en la base de datos de la aplicación	
Condiciones de Ejecución: La aplicación es ejecutada con privilegios de administración, los datos del profesor deben ser adicionados correctamente por el sistema.	
Entrada / Pasos de ejecución: Se intenta adicionar un profesor con sus datos	
Resultado Esperado: Los datos del profesor son adicionados correctamente en la base de datos de la aplicación.	
Evaluación de la Prueba: Prueba satisfactoria.	

Anexo II Tabla 4.7. 18 Pruebas de aceptación para la historia de usuario eliminar profesor.

Caso de Prueba de Aceptación	
Código: HU17_P1	Historia de Usuario: Eliminar profesor
Nombre: Comprobar que elimina profesor seleccionado	
Descripción: Probar que se elimina correctamente los datos del profesor en la base de datos de la aplicación	

<p>Condiciones de Ejecución: La aplicación es ejecutada con privilegios de administración, los datos del profesor deben ser eliminados correctamente por el sistema.</p>
<p>Entrada / Pasos de ejecución: Se intenta eliminar un profesor con sus datos de la base de datos</p>
<p>Resultado Esperado: Los datos del profesor son eliminados correctamente en la base de datos de la aplicación.</p>
<p>Evaluación de la Prueba: Prueba satisfactoria.</p>

Anexo II Tabla 4.7. 19 Pruebas de aceptación para la historia de usuario Listar grupos de quinto año.

Caso de Prueba de Aceptación	
Código: HU18_P1	Historia de Usuario: Listar grupos de quinto año.
Nombre: Comprobar el reporte listar grupos de quinto año.	
Descripción: Probar que se muestran correctamente los datos del reporte listar grupos de quinto año en la aplicación.	
<p>Condiciones de Ejecución: La aplicación es ejecutada por cualquier usuario del dominio, los datos del reporte listar grupos de quinto año deben ser mostrados correctamente por el sistema.</p>	
Entrada / Pasos de ejecución: Se intenta mostrar el reporte listar grupos de quinto año.	
Resultado Esperado: Los datos del reporte son mostrados correctamente en la aplicación.	
Evaluación de la Prueba: Prueba satisfactoria.	

Anexo II Tabla 4.7. 20 Pruebas de aceptación para la historia de usuario Adicionar estudiante

Caso de Prueba de Aceptación	
Código: HU20_P1	Historia de Usuario: Adicionar Estudiante
Nombre: Comprobar que adiciona profesor	

Descripción: Probar que se adiciona correctamente los datos del estudiante en la base de datos de la aplicación
Condiciones de Ejecución: La aplicación es ejecutada con privilegios de administración, los datos del estudiante deben ser adicionados correctamente por el sistema.
Entrada / Pasos de ejecución: Se intenta adicionar un estudiante con sus datos
Resultado Esperado: Los datos del estudiante son adicionados correctamente en la base de datos de la aplicación.
Evaluación de la Prueba: Prueba satisfactoria.

Anexo II Tabla 4.7. 21 Pruebas de aceptación para la historia de usuario Eliminar estudiante

Caso de Prueba de Aceptación	
Código: HU21_P1	Historia de Usuario: Eliminar estudiante
Nombre: Comprobar que elimina estudiante seleccionado	
Descripción: Probar que se elimina correctamente los datos del estudiante en la base de datos de la aplicación	
Condiciones de Ejecución: La aplicación es ejecutada con privilegios de administración, los datos del estudiante deben ser eliminados correctamente por el sistema.	
Entrada / Pasos de ejecución: Se intenta eliminar un estudiante con sus datos de la base de datos	
Resultado Esperado: Los datos del estudiante son eliminados correctamente en la base de datos de la aplicación.	
Evaluación de la Prueba: Prueba satisfactoria.	

Anexo II Tabla 4.7. 22 Pruebas de aceptación para la historia de usuario Listar estudiantes sin tesis.

Caso de Prueba de Aceptación	
Código: HU22_P1	Historia de Usuario: Listar estudiantes sin tesis

Nombre: Comprobar el reporte listar estudiantes sin tesis.
Descripción: Probar que se muestran correctamente los datos del reporte listar estudiantes sin tesis en la aplicación.
Condiciones de Ejecución: La aplicación es ejecutada por cualquier usuario del dominio, los datos del reporte listar estudiantes sin tesis deben ser mostrados correctamente por el sistema.
Entrada / Pasos de ejecución: Se intenta mostrar el reporte listar estudiantes sin tesis
Resultado Esperado: Los datos del reporte son mostrados correctamente en la aplicación.
Evaluación de la Prueba: Prueba satisfactoria.

Anexo III Opiniones del Usuario y del Tutor con respecto al trabajo.

OPINIÓN DEL USUARIO DEL TRABAJO DE DIPLOMA

El Trabajo de Diploma, titulado "Aplicación Web para el control del Proceso de Tesis de Grado en la Facultad 8", fue realizado en la Universidad de las Ciencias Informáticas. Esta entidad considera que, en correspondencia con los objetivos trazados, el trabajo realizado le satisface

- Totalmente
- Parcialmente en un ____ %

Los resultados de este Trabajo de Diploma le reportan a esta entidad los beneficios siguientes (cuantificar):

Y para que así conste, se firma la presente a los ____ días del mes de _____ del año _____.

Representante de la entidad

Cargo

Firma

Cuño

OPINIÓN DEL TUTOR DEL TRABAJO DE DIPLOMA

Título: "Aplicación Web para el control del Proceso de Tesis de Grado en la Facultad 8"

Autor: Reynaldo Heredia Rodríguez

Emilio Salcerio Paz

El tutor del presente Trabajo de Diploma considera que durante su ejecución los estudiantes mostraron las cualidades que a continuación se detallan.

Por todo lo anteriormente expresado considero que los estudiantes están aptos para ejercer como Ingenieros Informáticos; y propongo que se le otorgue al Trabajo de Diploma la calificación de

Yasim Mirabal Vetano

Firma

Fecha

Glosario de términos

PROYECTO

Es una empresa planificada que consiste en un conjunto de actividades que se encuentran interrelacionadas y coordinadas.

RELEASE

Versión de un producto informático.

MEGABYTE

(MB) es una unidad de medida de cantidad de datos informáticos. Es un múltiplo binario del octeto, que equivale a 106 (1.000.000 bytes) octetos, traducido a efectos prácticos como 106 (1.000.000) octetos.

XP

EXtreme Programming. Metodología de desarrollo de software basada en valores como simplicidad, comunicación, retroalimentación y coraje.

URL

Acrónimo de Universal Resource Locator (Localizador Universal de Recursos /Identificador Universal de Recursos). Sistema unificado de identificación de recursos en la red. Es el modo estándar de proporcionar la dirección de cualquier recurso en Internet.

HTTP

Hypertext Transfer Protocol. Protocolo de nivel de aplicación usado extensivamente en Internet para el acceso a documentos.

THHTTPD

HTTP daemon. Es un programa que corre de fondo en el servidor Web y espera peticiones externas al servidor. El daemon responde las peticiones y sirve los hipertextos y multimedia a través de Internet usando HTTP.

TCP/IP

Protocolo de control de transmisiones/Protocolo Internet. Es el protocolo estándar de comunicaciones en red utilizado para conectar sistemas informáticos a través de Internet.

WEBMAIL

Webmail es una tecnología que permite acceder a una cuenta de Correo Electrónico (E-Mail) a través de un navegador de Internet, de esta forma podrá acceder a su casilla de correo desde cualquier computadora del mundo.

LDAP

El protocolo LDAP (lightweight directory access protocol) es una versión "aligerada" del protocolo DAP, de modo que sea posible su utilización en redes TCP/IP.

APPLET

Un applet es un componente de una aplicación que se ejecuta en el contexto de otro programa, por ejemplo un navegador Web

GUI

Interfaz gráfica de usuario (Graphical User Interface). Entorno que muestra la información en pantalla en forma gráfica.

API

Interfaz de programación de aplicaciones (Applications Programming Interface): una serie de funciones que están disponibles para realizar programas para un cierto entorno.

CMS

Acrónimo de Content Management System, sistema de gestión o administración de contenidos

DRUPAL

Es un sistema de administración de contenido para sitios Web.

UML

(Unified Modeling Language) Lenguaje de modelado visual que se usa para especificar, visualizar, construir y documentar artefactos de un sistema de software.

LENGUAJE 4GL

(Fourth Generation Language). Lenguaje de cuarta generación. Lenguaje de programación que usa instrucciones de alto nivel para recuperar y dar forma a la información.

METALENGUAJE

Un metalenguaje es un lenguaje usado para hacer referencia a otros lenguajes. Los modelos formales de sintaxis para la descripción de la gramática, como por ejemplo, la gramática generativa, son tipos de metalenguaje.

OPEN SOURCE

Calificación de software que cumple una serie de requisitos, principalmente aquel que permite una libre redistribución, distribuye el código fuente, y permite modificaciones y trabajos derivados.

MULTIHILO

Una aplicación multihilo es aquella que ejecuta varias instrucciones a la vez, ideal para paralelizar procesos que pasan mucho tiempo en espera.

COMMIT

El comando COMMIT en SQL indica la finalización de una transacción de base de datos.