



Universidad de las Ciencias Informáticas

Facultad 1

Título: Sistema de Gestión de perfiles de tesis para la Facultad 1 de la Universidad de las Ciencias Informática.



Trabajo de Diploma para optar por el título de Ingeniero Informático

Autores: Leodan Cabrera Rodríguez

Tutor: Ing. César González Hernández

Co-Tutor: Ing. Adrian Antonio Garbey Fonseca

Enero 2011



"El conocimiento nos hace responsables."

Ernesto Che Guevara

DECLARACIÓN DE AUTORÍA

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

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

Leodan Cabrera Rodríguez

Autor

Ing. César González Hernández

Tutor

Ing. Adrian Antonio Garbey Fonseca

Co-Tutor

AGRADECIMIENTOS

A mi madre por estar siempre ahí para mí cuando la he necesitado, dándome sus consejos, comprendiendo mis problemas y enseñándome siempre el camino correcto.

A mi padre que a pesar de no estar ya junto a mí, fue y seguirá siendo siempre mi faro y guía en las cosas de la vida.

A mis hermanos, a todos por igual que siempre han estado ahí cuando les he necesitado con los cuales he compartido momentos de felicidad y también de tristeza, los que siempre me hacen pensar que la familia es lo primero.

A mi familia en general, a todos ellos que siempre me han apoyado en las decisiones que he tomado y que siempre han estado ahí en los momentos buenos pero también en los difíciles que es cuando más se les necesita.

A mi novia que a pesar de no estar aquí hoy, me dio fuerza en estos cinco años de la carrera para seguir adelante y por estar siempre a mi lado cuando la más la he necesitado.

A todos mis compañeros con los que compartido en estos cinco largos años, a los que aprecio mucho y con los cuales la estancia en esta escuela se me ha hecho más fácil.

A los profesores, que me han enseñado parte de lo que sé y en especial a mi tutor.

A la Revolución porque sin ella nada de estos que tenemos hoy, fuese realidad.

DEDICATORIA

A toda mi familia, a mi novia y a mis amigos por los cuales intento ser mejor cada día.

RESUMEN

En este trabajo se propone una solución basada en una aplicación web dinámica que automatice los procesos referentes a los Trabajos de Diplomas en la Facultad1 de la Universidad de la Ciencias Informáticas.

Entre las funcionalidades que presta el sistema se encuentra un banco de problemas que servirá de fuente generadora de temas para trabajos de diploma, la automatización del proceso de gestión y aprobación de los perfiles de trabajos de diploma, el proceso de revisión y registrar las notas de cada corte así como publicar la información referente a los cronogramas de discusión, la emisión de reportes estadísticos sobre la labor de gestión de las tesis, teniendo en cuenta los diferentes roles de los implicados en el sistema.

La aplicación es multiplataforma, cuenta con una interfaz amigable y fácil de usar lo que permite que un mayor número de usuarios puedan utilizar los servicios que brinda la misma. Para el desarrollo de la aplicación se utilizó Drupal como Sistema Gestor de Contenidos, PostgreSQL como Sistema Gestor de Base Datos y como lenguaje de programación PHP. Toda la documentación fue regida por la metodología SXP y como lenguaje de modelado UML.

TABLA DE CONTENIDO

Tabla de contenido

INTRODUCCIÓN	10
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA	14
1.1 INTRODUCCIÓN	14
1.2 PRINCIPALES CONCEPTOS ASOCIADOS AL DOMINIO DEL PROBLEMA.	14
1.2.1 Tesis.....	14
1.2.1.1 Objetivos fundamentales de las tesis.....	14
1.2.2 Automatización del proceso de gestión de los Trabajos de Diploma (TD)	15
1.2.3 ¿Qué es la Gestión?.....	15
1.2.3.1 Gestión del Conocimiento	15
1.2.3.1.1 Objetivos que han dado base a la Gestión del conocimiento.....	15
1.2.4 Las Bases de Datos	16
1.2.4.1 Principales características de los sistemas de base de datos.	16
1.2.5 Sistemas de Gestión de Bases de Datos	17
1.2.5.1 Ventajas de los SGBD	17
1.2.6 Aplicaciones Web	17
1.2.7 Lenguajes de programación.	17
1.3 SITUACIÓN INTERNACIONAL DE LOS SISTEMAS DE GESTIÓN	18
1.4 SITUACIÓN NACIONAL DE LOS SISTEMAS DE GESTIÓN	20
1.5 ANÁLISIS DE OTRAS SOLUCIONES EXISTENTES.	20
1.6 GESTOR DE BASES DE DATOS	21
1.6.1 MySQL	22
1.6.2 PostGreSQL.....	23
1.6.3 Fundamentación de la selección de PostgreSQL	24
1.7 SERVIDORES WEB	24
1.7.1 Apache	25
1.7.2 Thttpd.....	26
1.7.3 Roxen.....	26
1.7.3.1 Características más destacables.	26
1.7.3.2 Varias versiones 2.0.x de ROXEN son vulnerables a una serie de ataques.	27
1.7.4 Tcl Web Server.....	27
1.7.5 Fundamentación de la selección de Apache.....	28
1.8 METODOLOGÍAS DE DESARROLLO DE SOFTWARE Y LENGUAJE DE MODELADO	28
1.8.1 Metodologías Ágiles.....	28
1.8.1.1 SXP.....	29
1.8.2 Lenguaje Unificado de Modelado (UML).....	30
1.9 LENGUAJES DE PROGRAMACIÓN	31
1.9.1 PHP Hypertext Pre-processor (PHP)	31
1.9.1.1 Ventajas.....	31
1.10 HERRAMIENTAS DE DESARROLLO DE SOFTWARE (HDS)	32
1.10.1 Mejoras que provee el uso de HDS.....	33
1.10.2 Netbeans 6.9	33
1.10.2.1 Características de Netbeans 6.9	33
1.11 SISTEMAS GESTORES DE CONTENIDOS (CMS)	33
1.11.1 Drupal.....	34
1.12 FUNDAMENTACIÓN DE LA SELECCIÓN DE DRUPAL.....	36
1.13 CONCLUSIONES PARCIALES	36

TABLA DE CONTENIDOS

CAPÍTULO 2: PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA.....	37
2.1 INTRODUCCIÓN	37
2.2 DESCRIPCIÓN DEL PROBLEMA	37
2.3 OBJETO DE AUTOMATIZACIÓN.	37
2.4 PROPUESTA DEL SISTEMA	37
2.5 PROCESO DE DESARROLLO DEL PROYECTO BASADO EN SXP.	38
2.5.1 <i>Planificación del proyecto por roles</i>	38
2.6 MODELO DE DOMINIO	39
2.7 LISTA DE RESERVA DEL PRODUCTO (LRP).	40
2.8 HISTORIAS DE USUARIO.	41
<i>R1: Autenticar usuario</i>	41
Tareas de Ingeniería (R1).....	41
<i>R2: Gestionar pperfiles</i>	42
Tareas de Ingeniería (R2).....	42
<i>R3: Mostrar banco de perfiles</i>	43
Tareas de Ingeniería (R3).....	43
<i>R4: Revisar perfil</i>	44
Tareas de Ingeniería (R4).....	44
<i>R5: Mostrar cortes realizados</i>	45
Tareas de Ingeniería (R5).....	45
<i>R6: Asignar perfil de tesis a tesista</i>	45
Tareas de Ingeniería (R6).....	46
<i>R7: Gestionar tribunal de tesis</i>	46
Tareas de Ingeniería (R7).....	47
<i>R8: Asignar perfiles a tribunal de tesis</i>	48
Tareas de Ingeniería (R8).....	48
<i>R9: Crear acta de evaluación</i>	49
Tareas de Ingeniería (R9).....	49
<i>R10: Gestionar cronograma</i>	50
Tareas de Ingeniería (R10).....	50
<i>R11: Mostrar banco de perfiles reducidos</i>	51
Tareas de Ingeniería (R11).....	51
<i>R12: Gestionar centro</i>	51
Tareas de Ingeniería (R12).....	52
<i>R13: Gestionar departamento</i>	52
Tareas de Ingeniería (R13).....	53
<i>R14 Gestionar ffacultad</i>	53
Tareas de Ingeniería (R14).....	53
<i>R15 Gestionar ddatos del pprofesor</i>	54
Tareas de Ingeniería (R15).....	54
2.9 PLAN DE RELEASE	54
2.10 MODELO DE DATOS.	55
2.11 PATRONES Y ARQUITECTURA	56
2.12 SEGURIDAD EN DRUPAL	58
2.13 CONCLUSIONES PARCIALES.....	59
CAPÍTULO 3: CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA	60
3.1 INTRODUCCIÓN	60
3.2 DIAGRAMAS DE COMPONENTES.....	60

TABLA DE CONTENIDOS

3.3 DIAGRAMA DE PAQUETES.....	61
3.4 DIAGRAMA DE DESPLIEGUE	62



.....	62
3.5 MODELOS DE PRUEBA.....	62
<i>Nombre del caso de uso: Autenticar usuario.....</i>	<i>63</i>
<i>Nombre del caso de uso: Crear Perfil de Tesis.....</i>	<i>63</i>
<i>Nombre del caso de uso: Eliminar Perfil de Tesis.....</i>	<i>64</i>
<i>Nombre del caso de uso: Actualizar Perfil de Tesis.....</i>	<i>65</i>
<i>Nombre del caso de uso: Consultar Perfil de Tesis.....</i>	<i>65</i>
<i>Nombre del caso de uso: Mostrar banco de perfiles.....</i>	<i>66</i>
<i>Nombre del caso de uso: Revisar perfil.....</i>	<i>66</i>
<i>Nombre del caso de uso: Mostrar cortes realizados.....</i>	<i>67</i>
<i>Nombre del caso de uso: Crear Tribunal de Tesis.....</i>	<i>67</i>
<i>Nombre del caso de uso: Eliminar Tribunal de Tesis.....</i>	<i>68</i>
<i>Nombre del caso de uso: Consultar Tribunal de Tesis.....</i>	<i>69</i>
<i>Nombre del caso de uso: Actualizar Tribunal de Tesis.....</i>	<i>69</i>
<i>Nombre del caso de uso: Crear acta de evaluación.....</i>	<i>70</i>
3.6 CONCLUSIONES	70
CONCLUSIONES GENERALES	71
RECOMENDACIONES	72
REFERENCIAS BIBLIOGRÁFICAS	73
BIBLIOGRAFÍA.....	74
ANEXOS	75
ANEXO 1 CREAR ACTA DE EVALUACIÓN.....	75
ANEXO 2 CREAR PERFIL DE TRABAJO DE DIPLOMA	76
ANEXO 3 CREAR TRIBUNAL	77
ANEXO 4 TRIBUNAL 1.....	78

INTRODUCCIÓN

En la actualidad la sociedad está creada por organizaciones basadas en el aprendizaje, cuyo capital máspreciado es el ser humano, el mismo se sustenta en un desarrollo tecnológico sin precedentes. Con el transcurso del tiempo y el desarrollo obtenido a nivel mundial el volumen de información ha crecido considerablemente, por lo que su manipulación se ha vuelto compleja y el ser humano ha utilizado el avance tecnológico en función de procesar toda esta información.

En este contexto, debe entenderse que las tecnologías de información son un medio para transmitir y gestionar datos, información y conocimiento, el conocimiento es factor fundamental para la creación de riquezas.

El impacto de los cambios económicos y tecnológicos ha originado una revolución en materia de gestión de información. Se transformaron entonces las normas, los conceptos, los procedimientos, el comportamiento, así como los productos y los servicios. Una nueva actitud rige el quehacer cotidiano de la proyección y el desarrollo de las actividades de información. En este sentido, Gilberto Sotolongo expresa que: "la gestión de la información se ocupa de los resultados finales, no sólo de citas y localizaciones". Por otra parte coincidimos con Fernández Molina cuando corrobora que: "los profesionales de la información y sus instituciones son un factor indispensable para la permanencia de la organización en el mercado" [1].

La formación de las nuevas generaciones en el uso de las Tecnologías de la Información y las Comunicaciones (TIC) y el aprovechamiento de estas para aumentar la calidad de todo el proceso docente educativo son elementos que buscan asegurar el futuro de un país.

Cuba mantiene la idea de que a la sociedad le es necesario universalizar el conocimiento como una de las formas de alcanzar una mejor calidad de vida para todos los ciudadanos, sin distinción de edad ni condición social. Conceptualmente, la Informatización de la Sociedad se define en Cuba como el proceso de utilización ordenada y masiva de las TIC para satisfacer las necesidades de información y conocimiento de todas las personas y esferas de la sociedad [2].

La contribución del uso masivo de las tecnologías y el acceso a Internet puede ser una respuesta decisiva a la economía nacional para salir del subdesarrollo. La Industria Cubana del Software (ICSW) está llamada a convertirse en una significativa fuente de ingresos nacional, como resultado del correcto aprovechamiento de las ventajas del considerable capital humano disponible.

La Universidad de las Ciencias Informáticas (UCI) es un ejemplo de una institución impulsora de la industria del software en Cuba, esto se evidencia en las soluciones planificadas para incrementar el

INTRODUCCIÓN

impacto de la Universidad en la informatización del país y en las exportaciones. La UCI tiene como principal objetivo formar ingenieros en Ciencias Informáticas que cultiven los valores que caracterizan al ciudadano revolucionario cubano y que alcancen competencias, en su campo profesional, culminando con la realización de un Trabajo de Diploma (TD) con calidad.

El TD es un trabajo donde el estudiante tiene la oportunidad de reconocer la necesidad, relevancia y utilidad de su investigación, la cual concierne a un problema o conjunto de problemas en un área definida de la ciencia. Su elaboración permite al egresado el uso de los conocimientos, ideas, experiencias y teorías que fue adquiriendo en su formación. Los mismos son condecorados con este título una vez hayan discutido satisfactoriamente su TD.

El proceso de seguimiento y control a los TD comienza en quinto año donde cada departamento según las necesidades existentes en sus proyectos o en la facultad elabora los temas de tesis. Una vez estos temas estén creados, cada tutor con sus estudiantes son los responsables de elaborar sus perfiles de tesis, estos son llevados ante un comité de tesis el cual es el responsable de aprobar o no el perfil según el alcance que posee el mismo. El avance del TD es complejo y requiere del tiempo de los profesores que conforman el comité, los cuales no siempre están disponibles debido a que deben cumplir con otras tareas. Todos estos factores imposibilitan el seguimiento personalizado a cada tesis, se retrasa el proceso donde se definen los señalamientos, recomendaciones y notas, por ende se afecta directamente al estudiante en el desarrollo de su TD.

Haciendo un análisis a la situación planteada, se puede inferir como **problema científico** de la investigación: ¿Cómo facilitar la gestión de perfiles de tesis en la Facultad 1?

Para dar solución al problema científico planteado se traza el siguiente **objetivo general**: Desarrollar una aplicación web para mejorar y facilitar el proceso de gestión de perfiles de tesis en la Facultad1.

Para lograr el cumplimiento del objetivo general se trazan los siguientes **objetivos específicos**:

- Describir la necesidad de desarrollar un sistema para la gestión de perfiles de tesis en la Facultad 1.
- Examinar el estado del Arte referente a los sistemas de gestión de perfiles de tesis.
- Fundamentar el entorno tecnológico a utilizar.
- Implementar las funcionalidades del sistema.
- Realizar las pruebas funcionales del sistema.

INTRODUCCIÓN

El trabajo investigativo se fundamentó en el siguiente **objeto de estudio**: el proceso de gestión de perfiles de tesis.

El **campo de acción** que se establece es el proceso de gestión de perfiles de tesis en la Facultad 1.

La **idea a defender** que se crea como base de la investigación efectuada es: Con la realización de un sistema de gestión de perfiles de tesis la Facultad 1 podrá darle un seguimiento personalizado a cada tesis, garantizará la buena calidad y la satisfacción de todo el personal implicado, ganando así en rapidez, organización y eficiencia en el proceso.

Para dar solución al objetivo de la investigación se plantean las siguientes **tareas de investigación**:

- Fundamentación de la necesidad de desarrollar un sistema para la gestión de perfiles de tesis en la Facultad 1.
- Selección de las tecnologías necesarias para el desarrollo del sistema.
- Indagación del estado del Arte referente a los sistemas de gestión de perfiles de tesis.
- Fundamentación del entorno tecnológico a emplear.
- Descripción de las funcionalidades del sistema.
- Desarrollo del Modelo de Diseño del sistema.
- Implementación de las funcionalidades del sistema.
- Realización de las pruebas funcionales al sistema.

Un **método de investigación** provee estrategias elementales para ahorrar tiempo y esfuerzo en una investigación científica. Los métodos que se emplean para darle solución a estas tareas son:

Métodos Teóricos:

Análisis histórico-lógico: Este método lo podemos ver en el análisis de cómo los sistemas de gestión han venido comportándose históricamente, su gran importancia en la sociedad así como sus características, objetivos, entre otros aspectos. Además, se utilizó en el análisis del comportamiento del proceso de tesis que se ha realizado en la Facultad 1.

Analítico-sintético: Se emplea en el análisis y síntesis de la documentación utilizada para la bibliografía.

Métodos Empíricos

Observación: Se emplea para identificar algunas características en el proceso de gestión de tesis como

INTRODUCCIÓN

la forma de realización, quienes intervienen, que utilizan, entre otras.

Entrevista: Se realizaron entrevistas a profesores que interactúan directamente en el proceso de gestión de tesis, a profesores y estudiantes que estuvieron trabajando anteriormente en temas relacionados con la gestión de tesis para ayudar a identificar algunos de los subprocesos automatizables, y conocer el negocio.

Con el desarrollo de la investigación se espera obtener los siguientes **resultados**:

- Con esta tesis se espera obtener un sistema que gestione de una mejor forma y con una mayor facilidad el proceso de gestión de perfiles de tesis.
- Además se espera que este producto pueda ser utilizado no solo por la Facultad 1 sino por la UCI y todas la Universidades que puedan hacer uso de mismo de forma automatizada en los procesos de gestión de perfiles de tesis.

El trabajo cuenta de tres capítulos en el cual se abarca todo lo referente a la investigación realizada para alcanzar el éxito del objetivo trazado.

En el **Capítulo 1**: “Fundamentación teórica”, se relacionan conceptos asociados al dominio del problema y del campo de acción. Además se analizan las tendencias de las tecnologías y sistemas relacionados al tema investigado, así como las herramientas y tecnologías útiles para el desarrollo del sistema que se propone.

El **Capítulo 2**: “Presentación de la solución propuesta”, comprenderá la presentación del Modelo del negocio, los requerimientos y la descripción del sistema propuesto.

Ya en el **Capítulo 3**: “Construcción de la solución propuesta” se hace énfasis en la fase de análisis y diseño, implementación, posteriormente en la de prueba del sistema.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

1.1 Introducción

En la actualidad las tecnologías se encuentran en constante desarrollo y evolución. En este capítulo se realiza un estudio de las mismas para llevar a cabo la selección de las herramientas, el lenguaje de programación y la metodología de desarrollo con el propósito de dar solución al problema propuesto.

En el mismo se lleva a cabo un análisis de los principales conceptos y temas relacionados con la gestión de tesis y las tecnologías actuales, lo cual ayuda al mejor entendimiento del trabajo en general.

1.2 Principales conceptos asociados al dominio del problema.

Existe una serie de conceptos que resultan de vital importancia destacar. Conociéndolos se puede lograr un mejor entendimiento y dominio del campo de acción.

1.2.1 Tesis

Usualmente se realizan tesis a fin de obtener ciertos grados académicos, dando respuesta, a través de éstas, a ciertos problemas de investigación. De este modo, la tesis estaría comprendida por la respuesta que se le da a dicho problema, y no, como usualmente se piensa, a todo el texto, sin embargo, es así como se concibe en la mayoría del sector académico.

Como plantea Valarino una tesis es: “Un estudio que debe ser original, inédito, de naturaleza teórica, profesional, técnica, científica o humanística, en el cual se demuestran los conocimientos adquiridos y la habilidad para aplicarlos en la solución del problema planteado. Además debe ser expuesto sistemáticamente, representar un aporte al conocimiento de la disciplina, poseer una complementación bibliográfica actualizada y demostrar la independencia de criterio intelectual y científico de su autor y el dominio de los métodos de investigación propios del área del conocimiento respectivo” [3].

Su realización se debe hacer por lo general en un lapso establecido que puede variar en cada programa y finaliza con una defensa oral generalmente pública y coordinada por el jurado elegido para tal fin.

1.2.1.1 Objetivos fundamentales de las tesis.

Las tesis tienen varios objetivos fundamentales, entre los que se encuentran:

- Aplicación del conocimiento o creación artística para formular y solucionar problemas disciplinarios, interdisciplinarios, artísticos o profesionales.
- Desarrollar competencias en el estudiante en investigación.

- El manejo de instrumentos y los procesos de investigación y de creación.

1.2.2 Automatización del proceso de gestión de los Trabajos de Diploma (TD)

Con esta investigación se quiere lograr la automatización del proceso de gestión de los TD con un sistema informático. Para el análisis del concepto: la automatización, se coincide con Blanca Almazán cuando plantea que la automatización es: “Sistema de fabricación diseñado con el fin de usar la capacidad de las máquinas para llevar a cabo determinadas tareas anteriormente efectuadas por seres humanos, y para controlar la secuencia de las operaciones sin intervención humana” [4]. El término automatización es una amplia variedad de sistemas y procesos que operan con mínima o sin intervención del ser humano, ayudando a la liberación de los recursos humanos para que realicen tareas que requieran mayores conocimientos.

El desarrollo de la automatización comenzó a finales de la década de los 60 e inicios de los 70 con la aparición de los microordenadores. En todo este proceso se ha alcanzado una madurez en esta disciplina debido a la necesidad cada vez mayor de manejar y controlar grandes volúmenes de información, así como agilizar procesos y optimizar los recursos.

1.2.3 ¿Qué es la Gestión?

La gestión es un proceso que hace referencia a la acción y al efecto de gestionar o de administrar, este se realiza para mejorar la eficacia de las actividades que se desarrollan en torno al objeto de estudio. Por lo que: “Gestionar es realizar diligencias conducentes al logro de un negocio o de un deseo cualquiera” [5]. Tradicionalmente se han considerado cuatro funciones básicas en la gestión: planificación, organización, ejecución y control. Con la puesta en marcha de cada una de ellas se puede lograr una gestión efectiva.

1.2.3.1 Gestión del Conocimiento

La Gestión del Conocimiento es, la gestión de los activos intangibles que generan valor para la organización. La mayoría de estos intangibles tienen que ver con procesos relacionados de una u otra forma con la captación, estructuración y transmisión de conocimiento. Por lo tanto, la Gestión del Conocimiento tiene en el aprendizaje organizacional su principal herramienta. La Gestión del Conocimiento es un concepto dinámico [6].

1.2.3.1.1 Objetivos que han dado base a la Gestión del conocimiento.

- Formular una estrategia de alcance organizacional para el desarrollo, adquisición y aplicación del conocimiento.

- Implantar estrategias orientadas al conocimiento.
- Promover la mejora continua de los procesos de negocio, enfatizando la generación y utilización del conocimiento.
- Monitorear y evaluar los logros obtenidos mediante la aplicación del conocimiento.
- Reducir los tiempos de ciclos en el desarrollo de nuevos productos, mejoras de los ya existentes y la reducción del desarrollo de soluciones a los problemas.
- Reducir los costos asociados a la repetición de errores.

1.2.4 Las Bases de Datos

En un sistema de información los objetivos fundamentales no es sólo contar con recursos de información, sino también con los mecanismos necesarios para poder encontrar y recuperar estos recursos. De esta forma, las bases de datos se han convertido en un elemento indispensable para la creación de sedes web, Intranets y otros sistemas de información en los que se precisa manejar grandes o pequeños volúmenes de información.

Una base de datos es: “Una colección de datos organizados y estructurados según un determinado modelo de información que refleja no sólo los datos en sí mismos, sino también las relaciones que existen entre ellos. Una base de datos se diseña con un propósito específico y debe ser organizada con una lógica coherente. Los datos podrán ser compartidos por distintos usuarios y aplicaciones, pero deben conservar su integridad y seguridad al margen de las interacciones de ambos. La definición y descripción de los datos han de ser únicas para minimizar la redundancia y maximizar la independencia en su utilización” [7].

1.2.4.1 Principales características de los sistemas de base de datos.

- Independencia lógica y física de los datos.
- Redundancia mínima.
- Acceso concurrente por parte de múltiples usuarios.
- Integridad de los datos.
- Consultas complejas optimizadas.
- Seguridad de acceso y auditoría.
- Respaldo y recuperación.
- Acceso a través de lenguajes de programación estándar.

1.2.5 Sistemas de Gestión de Bases de Datos

Los Sistemas de Gestión de Bases de Datos (SGBD) son un tipo de software muy específico, dedicado a servir de interfaz entre la base de datos, el usuario y las aplicaciones que la utilizan. Es el software que permite a los usuarios procesar, describir, administrar y recuperar los datos almacenados en una base de datos.

Los SGBD tienen como propósito general manejar de manera clara, sencilla y ordenada un conjunto de datos que luego se convertirán en información relevante, para un buen manejo de los datos.

1.2.5.1 Ventajas de los SGBD

Es importante destacar que: Los SGBD ayudan a realizar las siguientes acciones:

- Control sobre la redundancia de datos.
- Consistencia de datos.
- Compartición de datos.
- Mayor eficiencia en la recogida, validación y entrada de datos.
- Facilidad para mantener los estándares establecidos.
- Mejora en el mantenimiento de la aplicación.
- Mejora en la seguridad.
- Mejora de accesibilidad de datos.

1.2.6 Aplicaciones Web

Una aplicación web es: “Un sistema que permite a un usuario final acceder a una parcela de información contenida en el universo al que hace referencia la definición del W3C. (Word Wide Web Consortium)” [8].

Las aplicaciones web son fragmentos de código que se ejecutan en el servidor Web cuando se realizan ciertas peticiones en el cliente web. El cliente Web es el encargado de ejecutarlas en la máquina del usuario. Son las aplicaciones tipo Java o Java Script. Mientras que el servidor Web es quien proporciona el código de las aplicaciones al cliente y éste, mediante el navegador, las ejecuta.

1.2.7 Lenguajes de programación.

Para la creación de cualquier sistema se hace uso de los lenguajes de programación. Estos son herramientas que nos permiten crear programas y software. Entre ellos tenemos Delphi, Visual Basic, Pascal, Java, entre otros. Los lenguajes de programación de una computadora en particular se conocen como código de máquinas o lenguaje de máquinas. El lenguaje máquina es el único que entiende

FUNDAMENTACIÓN TEÓRICA

directamente la computadora, ya que está escrito en lenguaje directamente entendible por la máquina, utiliza el alfabeto binario, que consta de los dos únicos símbolos 0 y 1, denominados bits.

Los lenguajes de programación facilitan la tarea de programación, ya que disponen de formas adecuadas que permiten ser leídas y escritas por personas, a su vez resultan independientes del modelo de computador a utilizar. Está constituido por un grupo de reglas gramaticales, un grupo de símbolos utilizables, un grupo de términos con sentido único y una regla principal que resume las demás. Los lenguajes pueden ser de alto o bajo nivel. En los de bajo nivel las instrucciones son simples y cercanas al funcionamiento de la máquina. En los lenguajes de alto nivel hay un alto grado de abstracción y el lenguaje es más próximo a los humanos.

Por lo tanto, un lenguaje de programación tiene varias ventajas:

- Es mucho más fácil de comprender que un lenguaje máquina.
- Permite mayor portabilidad, es decir que puede adaptarse fácilmente para ejecutarse en diferentes tipos de equipos.

1.3 Situación internacional de los sistemas de gestión

El estudio sobre el tema de la gestión de tesis electrónica permitió conocer que a nivel internacional existen varios sitios y aplicaciones que proponen soluciones basadas en la gestión de información que soportan el proceso de elaboración de tesis electrónicas. Entre ellos se pueden citar el trabajo titulado La Gestión Documental de las Tesis Doctorales en la Universidad de Castilla-La Mancha que consiste en una aplicación informática llamada TESIS desarrollada por los Servicios Informáticos de la Universidad de Castilla-La Mancha en el año 1998, a propuesta del Archivo General Universitario.

La aplicación permite realizar, compartida y de forma automatizada, la gestión administrativa de las tesis doctorales presentadas en la institución. Mediante TESIS se controla el trámite y la localización física de la tesis doctoral a lo largo de todo el procedimiento para su lectura, desde el trámite de presentación al archivo, incluyendo además otras posibilidades como es el análisis del contenido o la información relativa a la posterior publicación de estos trabajos de investigación.

Otro ejemplo es el trabajo de gestión de las tesis doctorales en la Universidad de Granada. La Universidad de Granada (UGR) cuenta desde 2005 con el Proyecto DIGIBUG, un modelo que mejora la visibilidad y accesibilidad de su producción científica, con el objetivo de acercarla a los investigadores, doctorandos, estudiantes y a la sociedad en general, ofreciendo una nueva perspectiva de difusión, utilización, citación y seguimiento de las diferentes líneas de investigación existentes en la UGR.

FUNDAMENTACIÓN TEÓRICA

El proyecto pretende abarcar la totalidad de la producción científica generada en el seno de la UGR. Uno de los materiales cuyo acceso y disponibilidad resultan más complejos son las tesis doctorales. La universidad contaba en aquel momento con una colección aproximada de 12.000 tesis, en soporte papel dispersas entre los distintos departamentos, bibliotecas y facultades, y cuyo acceso no era posible por estar ilocalizables.

Este sistema cuenta con las siguientes funcionalidades:

- Comunicar y difundir los resultados intelectuales a la comunidad científica.
- Facilitar el uso y accesibilidad tanto a usuarios internos como externos a la información científica.
- Maximizar la visibilidad, el uso y el impacto de la producción científica y académica de la UGR en la comunidad científica internacional.
- Retroalimentar la investigación.
- Apoyar las iniciativas de sus científicos y académicos.
- Incrementar su audiencia.
- Incrementar el impacto de los trabajos que desarrollan.
- Maximizar los rendimientos de la inversión, dado el presumible incremento del uso de las publicaciones.
- Maximizar la comprensión científica.

Aproximar la ciencia a la sociedad mediante el acceso abierto.

Un proyecto de este nivel no puede entenderse sin un espacio web de apoyo, permanente y actualizado, que sirva de guía a los doctorandos e investigadores de la UGR sobre los pasos que han de seguir para presentar sus tesis doctorales. La página de este sub-proyecto contiene la siguiente información:

- Procedimiento a seguir para depositar y consultar las tesis doctorales de la Universidad de Granada en exposición pública.
- Procedimiento a seguir para colgar y consultar una tesis leída en la Universidad de Granada.
- Tesis en formato electrónico leídas en departamentos de la Universidad de Granada.

1.4 Situación nacional de los sistemas de gestión

En la sociedad los procesos docentes educativos en los centros universitarios han adquirido gran importancia, lo cual ha provocado la existencia de sistemas o aplicaciones para la gestión y el control eficiente de los mismos. A continuación se hace un estudio crítico sobre los principales sistemas existentes en el ámbito nacional para el control de tesis.

En la Universidad de las Ciencias Informáticas podemos encontrar Tesis.uci.cu.

La gestión de las tesis en nuestro país está más vinculada con la gestión de la información. Existe en la Universidad de las Ciencias Informáticas un sitio web (Tesis.uci.cu) que tiene como objetivo brindar asesoría en los procesos de gerencia, desarrollo y tutoría de trabajos de grado, tesis y trabajos de investigación. Dicho sitio no cumple con los objetivos propuestos en este trabajo de diploma ni da solución a la problemática que se debe resolver.

Otro ejemplo que se puede ver es la aplicación Web Editorial Digital Universitaria de MOA, la cual se dedica a publicar libros, folletos, monografías, manuales, tutoriales y tesis relacionadas con las ciencias técnicas y humanidades en soporte digital. Tiene carácter no comercial, los documentos publicados están protegidos por licencia "Bienes Comunes Creativos de tipo Reconocimiento no Comercial sin Obra Derivada. Tiene como objetivo fundamental divulgar materiales de carácter científico y de apoyo al proceso docente universitario. Esta aplicación solamente permite almacenar las tesis y permitir la consulta de las mismas, pero no define entre sus funcionalidades los procesos internos de la gestión de tesis.

1.5 Análisis de otras soluciones existentes.

Es importante que todos los centros cuenten con un sistema de gestión, ya que permiten una mayor facilidad en el trabajo y la búsqueda de información necesaria. Luego de hacer un minucioso estudio preliminar del negocio en busca de sistemas que contribuyeran con la realización de la propuesta de solución planteada en el trabajo de diploma, se encontraron algunas propuestas que aunque no llegan a satisfacer las principales necesidades que se tenían para el desarrollo del sistema que se desea implementar, se decidió mostrar algunas de las características de ellos.

En la universidad se detectaron 4 soluciones anteriores. Las soluciones son las siguientes:

- Propuesta de diseño de una aplicación Web para la gestión de perfiles de los trabajos de diploma. (Carcassés; Estrada. 2007)

FUNDAMENTACIÓN TEÓRICA

- Aplicación Web para el control del proceso de tesis de grado en la facultad 8. (Heredia; Paz. 2008)
- Automatización de la gestión y reproducción de las tesis en la UCI. (Núñez. 2008)
- Sistema Gestor de Trabajos de Diploma.(Díaz;González.2009)

La solución 1 presenta los requerimientos para la gestión y revisión de los perfiles, la conformación de tribunales y comisiones de tutores, el registro y verificación de los usuarios y la elaboración de reportes. No se está utilizando en el proceso de tesis.

La solución 2 presenta los requerimientos para la autenticación y gestión de usuarios, la propuesta, revisión y rechazo de perfiles de tesis, la gestión de evaluaciones de las tesis y algunos reportes. El tutor del trabajo de diploma es el único que permanece en la universidad y no tiene consigo la solución.

La solución 3 presenta requerimientos para gestionar el banco de problemas, gestionar los perfiles, gestionar notas de cortes, impresión del documento, publicación del cronograma de discusión y el establecimiento del acceso a la aplicación. No se está usando en el proceso de tesis.

La solución 4 presenta algunos de los requerimientos necesarios para dar respuesta a la situación existente en la Facultad 1 pero está creada con tecnología privativa, además de que está compuesta por una aplicación de escritorio y una aplicación web, conformando entre las dos el sistema en general.

Se concluye que la solución más apropiada es el desarrollo de una solución propia para la Facultad 1 porque las ya existentes no cumplen las funcionalidades planteados por el cliente, no se encuentran en funcionamiento en estos momentos además de que están desarrolladas con tecnología privativa lo que va en contra del proceso de migración al software libre que se lleva a cabo en la Facultad 1 y en la universidad. El diseño de la misma debe permitir en su momento la generalización para su uso en el resto de las facultades de la Universidad.

1.6 Gestor de Bases de Datos

Un SGBD consiste en una colección de datos interrelacionados y un conjunto de programas para acceder a los mismos. Las principales funciones que debe cumplir, se relacionan con la creación y mantenimiento de la base de datos, el control de accesos, la manipulación de datos de acuerdo con las necesidades del usuario, el cumplimiento de las normas de tratamiento de datos, evitar redundancias y mantener la integridad.

Existen distintos gestores de base de datos los cuales están contenidos en dos grupos diferentes como por ejemplo:

SGBD libres

- MySQL
- PostgreSQL
- Apache Derby

SGBD no libres

- MySQL
- Microsoft SQL Server
- Oracle

1.6.1 MySQL

MySQL es un Sistema Gestor de Bases de Datos Relacionales (SGBDR) que opera a través de un intérprete de comandos. Los comandos de MySQL siguen el estándar de SQL, que se trata de un lenguaje normalizado para operar con Bases de Datos Relacionales. [9]

Características de MySQL:

- Está desarrollado en C/C++.
- Se distribuyen ejecutables para cerca de diecinueve plataformas diferentes.
- La API se encuentra disponible en C, C++, Eiffel, Java, Perl, PHP, Python, Ruby y TCL.
- Está optimizado para equipos de múltiples procesadores.
- Es muy destacable su velocidad de respuesta.
- Se puede utilizar como cliente-servidor o incrustado en aplicaciones.
- Cuenta con un rico conjunto de tipos de datos.
- Soporta múltiples métodos de almacenamiento de las tablas, con prestaciones y rendimiento diferentes para poder optimizar el SGBD a cada caso concreto.
- Su administración se basa en usuarios y privilegios.
- Sus opciones de conectividad abarcan TCP/IP, sockets UNIX y socketsNT, además de soportar completamente ODBC.
- Los mensajes de error pueden estar en español y hacer ordenaciones correctas con palabras acentuadas o con la letra 'ñ', además es altamente confiable en cuanto a estabilidad se refiere.[10]

Principales deficiencias:

- El soporte de transacciones o la integridad referencial (la gestión de claves foráneas) en MySQL está condicionado a un esquema de almacenamiento de tabla concreto.
- Si el usuario no va a usar transacciones, puede usar el esquema de almacenamiento “tradicional” (MyISAM) y obtendrá mayor rendimiento.
- Si su aplicación requiere transacciones, deberá usar el esquema que lo permite (InnoDB), sin ninguna otra restricción o implicación.
- No soporta procedimientos almacenados
- No incluye disparadores.
- No incluye vistas.
- No incluye características de objetos como tipos de datos estructurados definidos por el usuario, herencia etc.[10]

1.6.2 PostGreSQL

PostgreSQL es un Sistema de Gestión de Bases de Datos Objeto-Relacionales (ORDBMS) que ha sido desarrollado de varias formas desde la década de 1980.

PostgreSQL ofrece una potencia adicional y sustancial al incorporar clases, herencia, tipos y funciones; cuatro conceptos adicionales básicos en una forma en la que los usuarios puedan extender fácilmente su sistema. Otras características que aportan potencia y flexibilidad adicional son las restricciones (constraints), disparadores (triggers), reglas (rules) e integridad transaccional. [9]

Características de PostgreSQL:

- Está desarrollado en C, con herramientas como Yacc y Lex.
- La API de acceso al SGBD se encuentra disponible en C, C++, Java, Perl, PHP, Python y TCL, entre otros.
- Cuenta con un rico conjunto de tipos de datos, permitiendo además su extensión mediante tipos y operadores definidos y programados por el usuario.
- Su administración se basa en usuarios y privilegios.
- Sus opciones de conectividad abarcan TCP/IP, sockets Unix y sockets NT, además de soportar completamente ODBC.
- Los mensajes de error pueden estar en español y hacer ordenaciones correctas con palabras acentuadas o con la letra ‘ñ’.
- Es altamente confiable en cuanto a estabilidad se refiere.

FUNDAMENTACIÓN TEÓRICA

- Puede extenderse con librerías externas para soportar encriptación, búsquedas por similitud fonética.
- Control de concurrencia multiversión, lo que mejora sensiblemente las operaciones de bloqueo y transacciones en sistemas multiusuario.
- Soporte para vistas, claves foráneas, integridad referencial, disparadores, procedimientos almacenados, sub-consultas y casi todos los tipos y operadores soportados en SQL92 y SQL99.
- Implementación de algunas extensiones de orientación a objetos. En PostgreSQL es posible definir un nuevo tipo de tabla a partir de otra previamente definida. [10]

Principales limitaciones:

- Puntos de recuperación dentro de transacciones. Actualmente, las transacciones abortan completamente si se encuentra un fallo durante su ejecución. La definición de puntos de recuperación permitirá recuperar mejor transacciones complejas.
- No soporta tablespaces para definir donde almacenar la base de datos, el esquema, los índices.
- El soporte a orientación a objetos es una simple extensión que ofrece prestaciones como la herencia, no un soporte completo.[10]

1.6.3 Fundamentación de la selección de PostgreSQL

Con PostgreSQL, nadie puede demandarlo por violar acuerdos de licencia, puesto que no hay costo asociado a la licencia del software. Además de las ofertas de soporte, se cuenta con una comunidad importante de profesionales y entusiastas de los que su compañía puede obtener beneficios. Ha sido diseñado y creado para tener un mantenimiento y ajuste mucho menor que otros productos, conservando todas las características, estabilidad y rendimiento. Es extremadamente común que compañías reporten que PostgreSQL nunca ha presentado caídas en varios años de operación de alta actividad. Ni una sola vez. Simplemente funciona. El código fuente está disponible para todos sin costo. Si su equipo necesita extender o personalizar de alguna manera, pueden hacerlo con un mínimo esfuerzo, sin costos adicionales. Administra grandes cantidades de datos, el cual es considerado como una de las bases de datos de código abierto (Open Source) más avanzada del mundo. Con el empleo de este gestor se garantiza integridad de los datos y la velocidad de acceso y consultas a la base de datos. Son estas las características que nos han encaminado a utilizarlo para este proyecto.

1.7 Servidores Web

Un servidor web es un programa que sirve para atender y responder a las diferentes peticiones de los navegadores, proporcionando los recursos que soliciten usando el protocolo HTTP o el protocolo HTTPS

(la versión cifrada y autenticada). Un servidor web básico cuenta con un esquema de funcionamiento muy simple, basado en ejecutar infinitamente el siguiente bucle:

- ❖ Espera peticiones en el puerto TCP indicado (el estándar por defecto para HTTP es el 80).
- ❖ Recibe una petición.
- ❖ Busca el recurso.
- ❖ Envía el recurso utilizando la misma conexión por la que recibió petición.
- ❖ Vuelve al segundo punto [11].

1.7.1 Apache

Ahora te preguntará por qué esa popularidad de este software libre grandemente reconocido en muchos ámbitos empresariales y tecnológicos, pues aquí algunas razones:

- ❖ Corre en una multitud de Sistemas Operativos, lo que lo hace prácticamente universal.
- ❖ Apache es una tecnología gratuita de código fuente abierta. El hecho de ser gratuita es importante pero no tanto como que se trate de código fuente abierto. Esto le da una transparencia a este software de manera que si queremos ver qué es lo que estamos instalando como servidor, lo podemos saber, sin ningún secreto, sin ninguna puerta trasera.
- ❖ Apache es un servidor altamente configurable de diseño modular. Es muy sencillo ampliar las capacidades del servidor Web Apache. Actualmente existen muchos módulos para Apache que son adaptables a este, y están ahí para que los instalemos cuando los necesitemos. Otra cosa importante es que cualquiera que posea una experiencia decente en la programación de C o Perl puede escribir un módulo para realizar una función determinada.
- ❖ Apache trabaja con gran cantidad de Perl, PHP y otros lenguajes de script. Perl destaca en el mundo del script y Apache utiliza su parte del pastel de Perl tanto con soporte CGI como con soporte modo Perl. También trabaja con Java y páginas jsp. Teniendo todo el soporte que se necesita para tener páginas dinámicas.
- ❖ Apache te permite personalizar la respuesta ante los posibles errores que se puedan dar en el servidor. Es posible configurar Apache para que ejecute un determinado script cuando ocurra un error en concreto.
- ❖ Tiene una alta configuración en la creación y gestión de logs. Apache permite la creación de ficheros de log a medida del administrador, de este modo puedes tener un mayor control sobre lo que sucede en tu servidor.

1.7.2 Thttpd

Thttpd es un servidor web extremadamente pequeño, rápido, portable y seguro. Ofrece las mismas prestaciones que otros servidores, como Apache, aunque en situaciones de carga extrema su rendimiento es más alto.

Su utilidad como servidor de propósito general es escasa. Su uso primordial suele ser el de servidor rápido de contenido estático, a veces como soporte de servidores Apache para servir contenido binario de tipo estático, como imágenes, dejando para Apache las páginas dinámicas o las más complejas. Utilizado como complemento de Apache para servir contenido de tipo estático ha logrado reducir la carga del servidor principal a sólo una centésima parte.

1.7.3 Roxen

Es un servidor web de licencia GNU, desarrollado por un grupo sueco que después fundaría la empresa Roxen Internet Services. Roxen (que antes se llamó Spider y después, Spinner) destaca por su gran cantidad de funcionalidades. Este servidor, desarrollado en el lenguaje "Pike", ofrece cientos de módulos que permiten el desarrollo sencillo de sitios web muy ricos y dinámicos, sin más herramienta que el servidor Roxen.

1.7.3.1 Características más destacables.

- ❖ Multiplataforma: puede ejecutarse en Windows, Linux, MAC OS/X, Solaris.
- ❖ Código libre.
- ❖ Interfaz de administración basada en web, completa y fácil de usar.
- ❖ Soporte para gráficos integrado que posibilita, mediante etiquetas de RXML (la extensión de HTML propia de Roxen), la generación de imágenes, títulos, gráficos, etc.
- ❖ Acceso integrado a bases de datos, que permite el acceso a PostgreSQL, Oracle, MySQL, etc.
- ❖ MySQL integrada.
- ❖ Programación de servidor con PHP, RXML, Java, Perl y CGI.
- ❖ Soporte criptográfico fuerte.
- ❖ Arquitectura modular, que permite la carga y descarga de extensiones del servidor cuando está en marcha.
- ❖ Independencia con respecto a la plataforma en los módulos desarrollados por el propio usuario.

A mediados del año 2000, a causa de la aparición de la versión 2.0 de Roxen, que interrumpía la compatibilidad con las versiones anteriores, especialmente con la 1.3, la más usada, un grupo de desarrolladores, entre los cuales estaban algunos de los fundadores de Roxen, inició un proyecto que

partía de la versión 1.3 para desarrollar un servidor que mantuviera la compatibilidad con esta versión. Este servidor web se llamó Caudium. Actualmente las dos plataformas, Roxen y Caudium, gozan de muy buena salud, de buenas relaciones entre ellas (los desarrolladores se esfuerzan por mantener la compatibilidad entre las APIs de ambas) y cuentan con una base fiel de usuarios.

Roxen es uno de los pocos casos en los que un excelente producto no ha triunfado por completo, ya que siempre se ha visto eclipsado por Apache. El obstáculo más grande de Roxen era probablemente que el inglés era solamente una segunda lengua a la gente que la documentaba. Pero la fraseología puede tomar a veces un número de pasos para entender.

1.7.3.2 Varias versiones 2.0.x de ROXEN son vulnerables a una serie de ataques.

- ❖ Listar el contenido de cualquier directorio, aunque exista un fichero de índice en el mismo.
- ❖ Obtener el código fuente de los ficheros RXML, script PIKE, CGIs.
- ❖ Bajo ciertas circunstancias, acceder a ficheros protegidos por ficheros de configuración ".htaccess".
- ❖ El ataque se realiza utilizando un "%00" (un byte "nulo") en la URL. Por ejemplo, accediendo a "http://www.victima.com/%00/" se listará el contenido del directorio raíz del servidor web, aun existiendo un fichero índice.

1.7.4 Tcl Web Server

- ❖ El estándar de interfaz CGI es compatible. Sin embargo, hay una sobrecarga de alta en CGI por el costo de la reproducción un proceso externo para manejar cada solicitud. El servidor proporciona alternativas que se describen más adelante.
- ❖ Incluyen instalaciones con el apoyo para que pueda componer una página de varias partes, de forma estática. De nuevo, esto no siempre es el mejor enfoque, pero lo apoyamos.
- ❖ La autenticación básica. Este es el "codificado contraseña" estándar que es ampliamente utilizado a pesar de que es relativamente fácil de descubrir si usted puede ver los paquetes de red.
- ❖ Soporte para SSL se planifica y se incorporarán sobre la base de la demanda del cliente. SSL proporciona autenticación fuerte entre el cliente y el servidor, así como el cifrado de datos.
- ❖ Mapas de imagen. Los mapas de imagen son procesados directamente por el servidor para reducir la sobrecarga de procesamiento.
- ❖ Norma de registro y seguimiento de instalaciones.
- ❖ El servidor funciona muy bien con TclPro ActiveState depurador para que pueda depurar las aplicaciones web.
- ❖ Bibliotecas Tcl para facilitar la aplicación de la solicitud directa URL de los OEM.

1.7.5 Fundamentación de la selección de Apache

Apache está diseñado para ser un servidor web potente y flexible que pueda funcionar en la más amplia variedad de plataformas y entornos. Las diferentes plataformas y entornos, hacen que a menudo sean necesarias diferentes características o funcionalidades. Se ha adaptado siempre a una gran variedad de entornos a través de su diseño modular. Este diseño permite a los administradores de sitios web elegir qué características van a ser incluidas en el servidor seleccionando que módulos se van a cargar, ya sea al compilar o al ejecutar el servidor. Apache puede soportar de una forma más fácil y eficiente una amplia variedad de sistemas operativos. El servidor puede personalizarse mejor para las necesidades de cada sitio web. Es el servidor web hecho por excelencia, su configuración, robustez y estabilidad hacen que cada vez millones de servidores reiteren su confianza en este programa.

1.8 Metodologías de Desarrollo de Software y Lenguaje de modelado

Plattini [1996], llega a la definición de metodología de desarrollo como “un conjunto de procedimientos, técnicas, herramientas, y un soporte documental que ayuda a los desarrolladores a realizar nuevo software”. [14]

Sintetizando lo anterior se puede decir que una metodología representa el camino para desarrollar software de una manera sistemática. Las metodologías persiguen tres necesidades principales:

- ❖ Mejores aplicaciones, conducentes a una mejor calidad.
- ❖ Un proceso de desarrollo controlado.
- ❖ Un proceso normalizado en una organización, no dependiente del personal.

Los procesos se descomponen hasta el nivel de tareas o actividades elementales, donde cada tarea está identificada por un procedimiento que define la forma de llevarla a cabo. Para aplicar un procedimiento se pueden usar una o más técnicas, pudiendo ser gráficos con textos.

1.8.1 Metodologías Ágiles

Todo desarrollo de software tiene riesgos y es difícil de controlar, pero si no se utiliza una metodología de por medio, lo que se obtiene son clientes insatisfechos con el resultado y desarrolladores aún más insatisfechos. Las metodologías ágiles intentan evitar los tortuosos caminos de las metodologías tradicionales enfocándose en la gente y los resultados. Se basan en promover iteraciones en el desarrollo a lo largo de todo el ciclo de vida del proyecto, logrando que se minimicen los riesgos desarrollando software en cortos tiempo. El software desarrollado en una unidad de tiempo es llamado una iteración, la cual debe durar poco tiempo. Cada iteración del ciclo de vida incluye: planificación,

FUNDAMENTACIÓN TEÓRICA

análisis de requerimientos, diseño, codificación, revisión y documentación.

Las Metodologías Ágiles se basan en los siguientes principios:

- ❖ Realizar entregas cortas en el tiempo y continuas.
- ❖ Dar la bienvenida a los cambios.
- ❖ Entregas periódicas y frecuentes que funcionen.
- ❖ Los clientes forman parte del equipo de desarrollo.
- ❖ Equipo con individuos motivados. Darles para ello el ambiente, apoyo y confianza.
- ❖ La comunicación directa es el método más eficiente y efectivo para comunicar información dentro de un equipo de desarrollo. Intenta evitar el teléfono, correos electrónicos, fax, entre otros.
- ❖ La medida principal de progreso es el software que funciona.
- ❖ Desarrollo sostenible. Es indispensable que exista paz y armonía en el equipo para que el proyecto tenga éxito.
- ❖ Buen diseño y calidad técnica.
- ❖ La simplicidad es algo básico.
- ❖ Equipos auto-organizados.
- ❖ El equipo debe realizar reflexiones periódicamente para plantearse cómo llegar a ser más efectivo.

Debido a las grandes ventajas que proporcionan estas metodologías se propone para el desarrollo de este trabajo el uso de la metodología ágil SXP.

1.8.1.1 SXP

SXP es una metodología compuesta por las metodologías SCRUM y XP que ofrece una estrategia tecnológica, a partir de la introducción de procedimientos ágiles que permitan actualizar los procesos de software para el mejoramiento de la actividad productiva fomentando el desarrollo de la creatividad, aumentando el nivel de preocupación y responsabilidad de los miembros del equipo, ayudando al líder del proyecto a tener un mejor control del mismo.

SCRUM es una forma de gestionar un equipo de manera que trabaje de forma eficiente y de tener siempre medidos los progresos, de forma que sepamos por dónde andamos.

FUNDAMENTACIÓN TEÓRICA

XP más bien es una metodología encaminada para el desarrollo; 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.

Consta de 4 fases principales:

- **Planificación-Definición:** donde se establece la visión, se fijan las expectativas y se realiza el aseguramiento del financiamiento del proyecto.
- **Desarrollo:** es donde se realiza la implementación del sistema hasta que esté listo para ser entregado.
- **Entrega:** es la puesta en marcha.
- **Mantenimiento:** es la fase donde se realiza el soporte para el cliente.

De cada una de estas fases se realizan numerosas actividades tales como el levantamiento de requisitos, la priorización de la Lista de Reserva del Producto, definición de las Historias de Usuario, Diseño, Implementación, Pruebas, entre otras; de donde se generan artefactos para documentar todo el proceso. Las entregas son frecuentes, y existe una refactorización continua, lo que nos permite mejorar el diseño cada vez que se le añada una nueva funcionalidad.

SXP está especialmente indicada para proyectos de pequeños equipos de trabajo, rápido, cambio de requisitos o requisitos imprecisos, muy cambiantes, donde existe un alto riesgo técnico y se orienta a una entrega rápida de resultados y una alta flexibilidad.

Ayuda a que trabajen todos juntos, en la misma dirección, con un objetivo claro, permitiendo además seguir de forma clara el avance de las tareas a realizar, de forma que los jefes pueden ver día a día cómo progresa el trabajo. Por todas estas características y ventajas es que ha sido escogida las metodologías ágiles en particular SXP para la elaboración de este proyecto.

1.8.2 Lenguaje Unificado de Modelado (UML).

UML es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad. Es un lenguaje gráfico para visualizar, especificar, construir y documentar los artefactos de un sistema que involucra una gran cantidad de software. Está compuesto por diversos elementos gráficos que se combinan para conformar diagramas. Debido a que es un lenguaje, cuenta con reglas para combinar tales elementos. La finalidad de estos diagramas es presentar diversas perspectivas de un sistema, a las cuales se les conoce como modelo. Un modelo UML describe lo que supuestamente hará un sistema, pero no dice cómo implementar dicho sistema.

También intenta solucionar el problema de propiedad de código que se da con los desarrolladores, al

implementar un lenguaje de modelado común para todos los desarrollos se crea una documentación también común, que cualquier desarrollador con conocimientos del mismo será capaz de entender, independientemente del lenguaje utilizado para el desarrollo.

Es importante recalcar que no es una guía para realizar el análisis y diseño orientado a objetos, es decir, no es un proceso, es un lenguaje que permite la modelación de sistemas con tecnología orientada a objetos y que es para especificar y no para describir métodos o procesos.

UML es ahora un estándar, no existe otra especificación de diseño orientado a objetos, ya que es el resultado de las tres opciones existentes en el mercado. Su utilización es independiente del lenguaje de programación y de las características de los proyectos, ya que UML ha sido diseñado para modelar cualquier tipo de proyecto.

1.9 Lenguajes de Programación

Un lenguaje de programación es un modo práctico para que los seres humanos puedan dar instrucciones a un equipo. Estos son utilizados para controlar el comportamiento físico y lógico de una máquina. Un lenguaje de programación permite a uno o más programadores especificar de manera precisa sobre qué datos debe operar una computadora, cómo estos datos deben ser almacenados o transmitidos y qué acciones debe tomar bajo una variada gama de circunstancias.

Para la confección de este sistema se utilizó el lenguaje de programación PHP.

1.9.1 PHP Hypertext Pre-processor (PHP)

PHP es un lenguaje interpretado de propósito general ampliamente usado y que está diseñado especialmente para desarrollo web y puede ser incrustado dentro de código HTML. Generalmente se ejecuta en un servidor web, tomando el código en PHP como su entrada y creando páginas web como salida. Puede ser desplegado en la mayoría de los servidores web y en casi todos los sistemas operativos y plataformas sin costo alguno.

1.9.1.1 Ventajas

- ❖ Es un lenguaje multiplataforma.
- ❖ Completamente orientado a la web.
- ❖ Capacidad de conexión con la mayoría de los motores de base de datos que se utilizan en la actualidad, destaca su conectividad con MySQL y PostgreSQL.
- ❖ 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.
- ❖ Es libre, por lo que se presenta como una alternativa de fácil acceso para todos. PHP es Open Source, lo cual significa que el usuario no depende de una compañía específica para arreglar cosas que no funcionan, además no estás forzado a pagar actualizaciones anuales para tener una versión que funcione.
- ❖ Permite las técnicas de Programación Orientada a Objetos.
- ❖ Biblioteca nativa de funciones sumamente amplia e incluida.
- ❖ No requiere definición de tipos de variables aunque sus variables se pueden evaluar también por el tipo que estén manejando en tiempo de ejecución.
- ❖ Tiene manejo de excepciones (desde PHP5).
- ❖ Rapidez. PHP generalmente es utilizado como módulo de Apache, lo que lo hace extremadamente veloz. Está completamente escrito en C, así que se ejecuta rápidamente utilizando poca memoria.
- ❖ Puede interactuar con muchos motores de bases de datos tales como MySQL, MS SQL, Oracle, Informix, PostgreSQL, y otros muchos. Siempre podrás disponer de ODBC para situaciones que lo requieran.

PHP se convierte en un buen candidato para la implementación del sistema que se propone teniendo en cuenta estas características y ventajas. Los desarrolladores han estudiado varios lenguajes y poseen mayor experiencia en el lenguaje PHP, por tanto se gana en el costo beneficio/tiempo al seleccionarlo para la construcción del sistema.

1.10 Herramientas de desarrollo de software (HDS)

Actualmente se considera a las HDS como herramientas basadas en computadoras que asisten el proceso de ciclo de vida de software, consolidadas en la literatura en la forma de Ingeniería de software asistida por computadora (CASE, por sus siglas en inglés). Esto es, software que se utiliza para ayudar a las actividades del proceso de software o software que es utilizado para diseñar y para implementar otro software. Permiten automatizar acciones bien definidas, reduciendo también la carga cognitiva del ingeniero de software, quien requiere libertad para concentrarse en los aspectos creativos del proceso. Este soporte se traduce en mejoras a la calidad y la productividad en el diseño y desarrollo.

1.10.1 Mejoras que provee el uso de HDS

- ❖ Apoyan a las metodologías y métodos, integrando actividades y propiciando visión de continuidad entre fases metodológicas.
- ❖ Mejoran la comunicación entre los actores involucrados, facilitándoles compartir su trabajo y desempeñarlo de forma dinámica e iterativa.
- ❖ Establecen métodos efectivos para almacenar y utilizar los datos, lo que permite organizar y relacionar componentes, para establecer acceso través de un repositorio.
- ❖ Agregan eficiencia al mantenimiento, ya que los programas son construidos sobre las mismas estructuras y estándares, facilitando la adherencia a la disciplina de diseño y facilitan también la conversión automática de programas a versiones más recientes de lenguajes de programación.
- ❖ Automatizan porciones del análisis y diseño tediosos y propensos a error, con influencia sobre la generación de código, las pruebas y el control. Resalta la consideración de que los beneficios potenciales sólo pueden ser alcanzados si las HDS son utilizadas de forma correcta.

1.10.2 Netbeans 6.9

El IDE Netbeans es una herramienta para programadores pensada para escribir, compilar, depurar y ejecutar programas. Está escrito en Java - pero puede servir para cualquier otro lenguaje de programación. Existe además un número importante de módulos para extender el IDE Netbeans. El IDE Netbeans es un producto libre y gratuito sin restricciones de uso.

1.10.2.1 Características de Netbeans 6.9

- ❖ Refactor y búsqueda de usos para CSS y lenguajes tipo HTML
- ❖ Auto-completado de código y links para atributos de CSS
- ❖ Refactor para extraer estilos inline en un estilo separado
- ❖ Soporte para PHP Zend Framework
- ❖ Nuevo formateador con muchas más reglas
- ❖ Más ayudas en el editor
- ❖ Soporte para comprobación de ortografía en el editor
- ❖ Se agregó una opción para reportar bugs directamente desde el IDE
- ❖ Soporte para múltiples servidores.

1.11 Sistemas Gestores de Contenidos (CMS)

Un Sistema de gestión de contenidos (Content Management System, en inglés, abreviado CMS) permite la creación y administración de contenidos principalmente en páginas web. Consiste en una interfaz que

controla una o varias bases de datos donde se aloja el contenido del sitio. El sistema permite manejar de manera independiente el contenido y el diseño. Así, es posible manejar el contenido y darle en cualquier momento un diseño distinto al sitio sin tener que darle formato al contenido de nuevo, además de permitir la fácil y controlada publicación en el sitio a varios editores. Un ejemplo clásico es el de editores que cargan el contenido al sistema y otro de nivel superior que permite que estos contenidos sean visibles a todo público.

1.11.1 Drupal

Drupal es un marco de administración de contenidos, sistema de administración de contenidos y motor de blogging basado en PHP, que fue originalmente escrito por Dries Buytaert y es el software usado para impulsar los sitios web Debian Planet, Terminus1525 , Spread Firefox y Kernel Trap.

Características de Drupal:

- ❖ **Ayuda on-line:** Un robusto sistema de ayuda online y páginas de ayuda para administradores.
- ❖ **Búsqueda:** Todo el contenido en Drupal es totalmente indexado en tiempo real y se puede consultar en cualquier momento.
- ❖ **Código abierto:** El código fuente de Drupal está libremente disponible bajo los términos de la licencia GNU/GPL.
- ❖ **Módulos:** La comunidad de Drupal ha contribuido muchos módulos que proporcionan casi todas las funcionalidades.
- ❖ **Personalización:** Un robusto entorno de personalización está implementado en el núcleo de Drupal.
- ❖ **URLs amigables:** Drupal usa el mod_rewrite de Apache para crear URLs que son manejables por los usuarios y los motores de búsqueda.
- ❖ **Autenticación de usuarios:** Los usuarios se pueden registrar e iniciar sesión de forma local o utilizando usuario y contraseña, para su uso en una intranet, Drupal se puede integrar con un servidor LDAP.

- ❖ **Permisos basados en roles:** Los permisos son asignados a roles, y los usuarios se agrupan en diferentes roles.
- ❖ **Control de versiones:** El sistema de control de versiones de Drupal permite seguir y auditar totalmente las sucesivas actualizaciones del contenido: qué se ha cambiado, la hora y la fecha, quién lo ha cambiado, y más. También permite mantener comentarios sobre los sucesivos cambios o deshacer los cambios recuperando una versión anterior.

- ❖ **Enlaces permanentes (Permalinks):** Todo el contenido creado en Drupal tiene un enlace permanente asociado a él para que pueda ser enlazado externamente sin temor de que el enlace falle en el futuro.
- ❖ **Objetos de Contenido (Nodos):** El contenido creado en Drupal es, funcionalmente, un objeto (Nodo). Esto permite un tratamiento uniforme de la información, como una misma cola de moderación para envíos de diferentes tipos, promocionar cualquiera de estos objetos a la página principal o permitir comentarios -o no- sobre cada objeto.
- ❖ **Plantillas (Templates):** El sistema de temas de Drupal separa el contenido de la presentación permitiendo controlar o cambiar fácilmente el aspecto del sitio web. Se pueden crear plantillas con HTML y/o con PHP.
- ❖ **Sindicación del contenido:** Drupal exporta el contenido en formato RDF/RSS para ser utilizado por otros sitios web.
- ❖ **Independencia de la base de datos:** Aunque la mayor parte de las instalaciones de Drupal utilizan MySQL, existen otras opciones. Drupal incorpora una 'capa de abstracción de base de datos' que actualmente está implementada y mantenida para MySQL y PostgreSQL, aunque permite incorporar fácilmente soporte para otras bases de datos.
- ❖ **Multiplataforma:** Drupal ha sido diseñado desde el principio para ser multi-plataforma. Puede funcionar con Apache o Microsoft IIS como servidor web y en sistemas como Linux, BSD, Solaris, Windows y Mac OS X. Por otro lado, al estar implementado en PHP, es totalmente portable.
- ❖ **Múltiples idiomas y Localización:** Drupal está pensado para una audiencia internacional y proporciona opciones para crear un portal multilingüe. Todo el texto puede ser fácilmente traducido utilizando una interfaz web, importando traducciones existentes o integrando otras herramientas de traducción como GNU ettext.
- ❖ **Administración vía Web:** La administración y configuración del sistema se puede realizar enteramente con un navegador y no precisa de ningún software adicional.
- ❖ **Registros e Informes:** Toda la actividad y los sucesos del sistema son capturados en un 'registro de eventos', que puede ser visualizado por un administrador.
- ❖ **Encuestas:** Drupal incluye un módulo que permite a los administradores y/o usuarios crear encuestas on-line totalmente configurables.
- ❖ **Foros de discusión:** Drupal incorpora foros de discusión para crear sitios comunitarios vivos y dinámicos.
- ❖ **Control de congestión:** Drupal incorpora un mecanismo de control de congestión que permite habilitar y deshabilitar determinados módulos o bloques dependiendo de la carga del servidor. Este mecanismo es totalmente configurable y ajustable.

- ❖ **Sistema de Cache:** El mecanismo de cache elimina consultas a la base de datos incrementando el rendimiento y reduciendo la carga del servidor.

1.12 Fundamentación de la selección de Drupal

Drupal es un sistema de gestión de contenido modular y muy configurable. Es un programa de código abierto, multiplataforma, multiusuario, multilenguaje y extensible con licencia GNU/GPL, escrito en PHP, desarrollado y mantenido por una activa comunidad de usuarios. Destacada por la calidad de su código y de las páginas generadas, el respeto de los estándares de la web, y un énfasis especial en la usabilidad y consistencia de todo el sistema. El diseño de Drupal es especialmente idóneo para construir y gestionar comunidades de Internet. No obstante, su flexibilidad y adaptabilidad, así como la gran cantidad de módulos adicionales disponibles, hace que sea adecuado para realizar muchos tipos diferentes de sitio web.

1.13 Conclusiones Parciales

En este capítulo se realiza un estudio de las tendencias actuales, se analizan los conceptos fundamentales relacionados a los sistemas de gestión y se propone la solución que más se puede ajustar a el desarrollo de este sistema. Se determinan las tecnologías y herramientas que se usarán. Llegando a la conclusión de que para desarrollar la tesis la metodología ágil usada será SXP con UML como lenguaje de modelado. Como gestor de Bases de Datos PostgreSQL para almacenar adecuadamente la información generada. Se utilizará Netbeans como herramienta de desarrollo y Visual Paradigm for UML como herramienta de modelado. Como lenguaje de programación PHP.

CAPÍTULO 2: PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA

2.1 Introducción

En el presente capítulo se realiza un análisis detallado de la solución propuesta para satisfacer las necesidades del cliente lo cual es el objetivo fundamental de todo producto de software. Se realiza una descripción del problema y se identifican los requerimientos funcionales y no funcionales que debe tener el sistema para poder conseguir una comprensión más precisa de los mismos.

2.2 Descripción del problema

En la Universidad de las Ciencias Informáticas las facultades tienen el papel principal a la hora de dirigir y controlar los procesos de docencia y producción. Gran cantidad de estos procesos orientados al cumplimiento de la preparación de los estudiantes se realizan de forma manual, sobre todo los que tienen lugar en los órganos de Organización y Gestión de Trabajos de Diploma. Por el grado de importancia que tiene esta tarea se hace necesaria la utilización de herramientas automatizadas que faciliten el flujo de información haciendo su organización y transmisión más rápida y eficaz. 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 Facultad¹. Por esta razón se desea construir una aplicación Web para automatizar estas tareas.

2.3 Objeto de automatización.

En toda la etapa de conformación y desarrollo de las tesis, desde la elaboración del perfil hasta la aprobación y discusión final existen varios procesos que deben ser automatizados, debido a que su ejecución de forma manual resulta más engorrosa y propensa a errores. Incluyéndole a esto el consumo de tiempo al personal implicado en la realización de las tareas. Por lo que se desea desarrollar una aplicación capaz de automatizar todo el proceso que involucran las tesis de la Facultad 1. Donde se espera lograr el control y centralización de los documentos concernientes al proceso de tesis, así como brindar los reportes necesarios para mantener un conocimiento general del estado de las mismas y del personal involucrado.

2.4 Propuesta del Sistema

Para dar solución al problema existente se ha decidido desarrollar una aplicación Web que facilite el trabajo con la información referente a las tesis, lo que nos podrá garantizar una mejor organización y coordinación de las mismas. Para desarrollar esta propuesta la aplicación se realizará utilizando el CMS Drupal con el cual se implementará una aplicación que automatizará el proceso de tesis de grado, cumpliendo los requisitos pedidos por el usuario. Los procesos fundamentales en los que estará

PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA

enmarcado el trabajo son: publicar los perfiles de tesis propuestos por los profesores, la aprobación de los perfiles de tesis, la asignación de los perfiles a los estudiantes de quinto año que aún no tienen uno asignado, tener un seguimiento de la evaluación de las tesis así como generar reportes estadísticos de cada corte por individual. El sistema quedará elaborado de forma tal que muestre al usuario solo aquellas opciones a la que está autorizado acceder y realizar en las interfaces solo los cambios que le sean permitidos dependiendo de las responsabilidades que posea. Se desea lograr un sistema que aporte soluciones reales, palpables, eficientes y satisfactorias a las necesidades planteadas.

2.5 Proceso de desarrollo del proyecto basado en SXP.

El objetivo de un proceso de desarrollo es subir la calidad del software a través de una mayor transparencia y control sobre el proceso. Con este objetivo y de forma organizada y efectiva, se toma como base en la metodología ágil SXP la creación artefactos (tabla de roles, historias de usuario, modelos auxiliares) que permiten un desarrollo centrado en sub-tareas, permitiendo mayor rapidez en el cumplimiento de las tareas asignadas.

2.5.1 Planificación del proyecto por roles

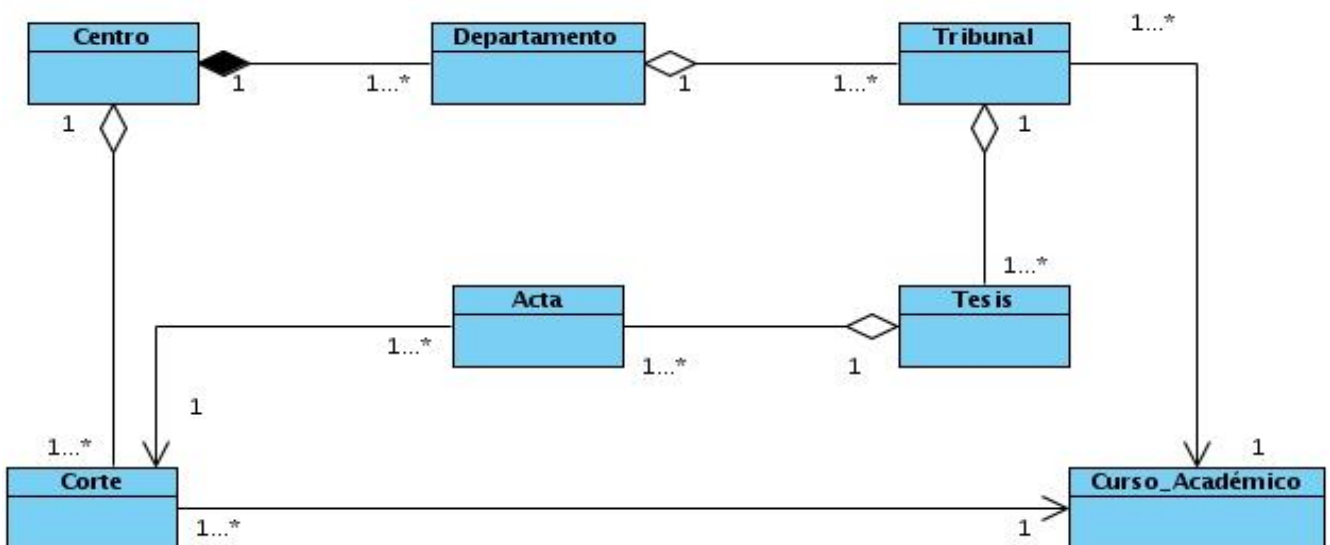
Rol	Responsabilidad	Nombre
Gerente	Dirige y controla las tareas del equipo. Toma las decisiones finales. Participa en la selección de objetivos y requerimientos. Controla el progreso y da seguimiento a cada iteración. Evalúa si los objetivos son alcanzables con las restricciones de tiempo y recursos presentes.	César González Hernández
Diseñadores	Encargados del diseño del sistema; máximos responsable de supervisar el proceso de construcción.	Leticia Rodríguez Viera y Leodan Cabrera Rodríguez
Cliente	Participa en las tareas que involucran la lista de reserva del producto. Contribuye a definir las historias de usuario y los casos de prueba de aceptación, para validar su implementación, y controla el progreso del software.	Vice-Decano de formación de la Facultad 1 y Subdirector de GEITEL. Centro Gestión de Información y Tecnologías.

PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA

Programador	Elabora el código de las nuevas funcionalidades a implementar. Escribe las pruebas unitarias. Debe existir una comunicación y coordinación adecuada entre los programadores y el resto del equipo.	Leodan Cabrera Rodríguez
Analista	Es el encargado de escribir las historias de usuario y las pruebas funcionales para validar su implementación.	Leticia Rodríguez Viera

2.6 Modelo de dominio

Un sistema por pequeño que parezca no lo hace sencillo, por ello se establece una técnica para la especificación de los requisitos más importantes del sistema “ El modelo de dominio” este puede utilizarse para capturar y expresar el entendimiento ganado en un área bajo análisis como paso previo al diseño de un sistema, ya sea de software o de otro tipo. El modelo de dominio puede ser tomado como el punto de partida para el diseño del sistema.



PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA

2.7 Lista de Reserva del Producto (LRP).

La Lista de Reserva del Producto (LRP) está conformada por una lista priorizada que define el trabajo a realizar en el proyecto. Tiene como objetivo comprobar que el producto resultante sea el definido, el más correcto, de utilidad y competitivo.

Prioridad	Ítem *	Descripción	Estimación	Estimado por
Muy Alta				
	1	Autenticar usuario	0.5	PROG
	2	Gestionar Perfiles	2	PROG
	3	Mostrar banco de perfiles	0.3	PROG
	4	Gestionar perfil de tesis a tesista.	0.2	PROG
	5	Mostrar perfiles reducidos	0.3	PROG
Alta				
	6	Gestionar tribunal de tesis	0.5	PROG
	7	Gestionar perfiles a tribunal de tesis.	1	PROG
	8	Revisar perfil	0.8	PROG
	9	Mostrar cortes realizados	0.6	PROG
	10	Gestionar centro	0.5	PROG
	11	Crear datos de acta de corte de tesis	1	PROG
	12	Gestionar Facultad	0.4	PROG
	13	Gestionar Datos del Profesor	0.3	PROG
	14	Gestionar Departamento	0.5	PROG
Media				
	15	Gestionar cronograma	0.7	PROG

PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA

2.8 Historias de usuario.

Una historia de usuario es una representación de un requerimiento de software escrito en una o dos frases utilizando el lenguaje común del usuario. Cuando se hace uso de la metodología SXP, para especificar los casos de uso como correspondería en RUP, se hace mediante historias de usuarios, descriptoras de las tareas que el sistema debe hacer, cuestión que depende en gran medida de las especificaciones realizadas por el cliente.

R1: Autenticar usuario

Historia de Usuario	
Número: HU-1	Nombre Historia de Usuario: Autenticar usuario
Modificación de Historia de Usuario Número: 1	
Usuario: Leodan Cabrera Rodríguez y Leticia Rodríguez Viera	Iteración Asignada: 1
Prioridad en Negocio: Muy Alta	Puntos Estimados: 0.5
Riesgo en Desarrollo: Bajo	Puntos Reales: 0.5
Descripción: Se produce cuando el usuario realiza la petición de conexión a la interfaz principal a partir de su usuario y contraseña. Cuando se introducen correctamente los datos el sistema carga y da acceso a la página principal.	
Observaciones: Para acceder a la interfaz del sistema, el usuario debe haberse autenticado con su usuario y contraseña del dominio UCI.	

Tareas de Ingeniería (R1)

Tarea de Ingeniería	
Número Tarea: 1	Número Historia de Usuario: HU-1
Nombre de la Tarea: Instalar y configurar el módulo ldap_integration.	
Tipo de Tarea: Programación	Puntos estimados: 0.3
Fecha Inicio: 19/1/2011	Fecha Fin: 19/1/2011
Programador Responsable: Leodan Cabrera y Leticia Rodríguez	

PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA

Descripción: Se instala el módulo y luego en la configuración del sitio en el menú LDAP se procede a la configuración del mismo. Para que el módulo funcione tiene que estar instalado el paquete PHP5-LDAP que se encuentra en el repositorio.

R2: Gestionar pperfiles

Historia de Usuario	
Número: HU-2	Nombre Historia de Usuario: Gestionar perfiles
Modificación de Historia de Usuario Número: 1	
Usuario: Leodan Cabrera Rodríguez	Iteración Asignada: 1
Prioridad en Negocio: Muy Alta	Puntos Estimados: 2
Riesgo en Desarrollo: Bajo	Puntos Reales: 1
Descripción: Este requisito se produce cuando un usuario crea un perfil con todos los datos que requiere, además cuando obtiene o modificar los datos del perfil de tesis.	
Observaciones: Para gestionar un perfil el usuario debe haberse autenticado como un profesor con categoría.	

Tareas de Ingeniería (R2)

Tarea de Ingeniería	
Número Tarea: 1	Número Historia de Usuario: HU-2
Nombre de la Tarea: Instalar el módulo CCK y crear el tipo de contenido perfil.	
Tipo de Tarea: Programación	Puntos estimados: 2
Fecha Inicio: 19/1/2011	Fecha Fin: 21/1/2011
Programador Responsable: Leodan Cabrera y Leticia Rodríguez	
Descripción: Se instala el módulo CCK el cual permite crear tipos de contenidos.	

Tarea de Ingeniería	
Número Tarea: 2	Número Historia de Usuario: HU-2

PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA

Nombre de la Tarea: Crear vistas para mostrar los perfiles.	
Tipo de Tarea: Programación	Puntos estimados:
Fecha Inicio: 21/1/2011	Fecha Fin: 23/1/2011
Programador Responsable: Leodan Cabrera y Leticia Rodríguez	
Descripción: Se instala el módulo views y viewsphfilter los cuales permiten crear vistas de los tipos de contenidos.	

R3: Mostrar banco de perfiles

Historia de Usuario	
Número: HU-3	Nombre Historia de Usuario: Mostrar banco de perfiles
Modificación de Historia de Usuario Número: 1	
Usuario: Leodan Cabrera Rodríguez	Iteración Asignada: 1
Prioridad en Negocio: Muy Alta	Puntos Estimados: 0.3
Riesgo en Desarrollo: Bajo	Puntos Reales: 0.3
Descripción: Este requisito se produce cuando un usuario autenticado y con los permisos asignados desea consultar el banco de perfiles.	
Observaciones:	

Tareas de Ingeniería (R3)

Tarea de Ingeniería	
Número Tarea: 1	Número Historia de Usuario: HU-3
Nombre de la Tarea: Crear una vista para mostrar el banco de perfiles.	
Tipo de Tarea: Programación	Puntos estimados:
Fecha Inicio: 25/1/2011	Fecha Fin: 25/1/2011
Programador Responsable: Leodan Cabrera y Leticia Rodríguez	
Descripción: A través de una vista que se crea se muestra el banco de perfiles.	

PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA

R4: Revisar perfil

Historia de Usuario	
Número: HU-4	Nombre Historia de Usuario: Revisar perfil
Modificación de Historia de Usuario Número: 1	
Usuario: Leodan Cabrera Rodríguez	Iteración Asignada: 2
Prioridad en Negocio: Alta	Puntos Estimados: 0.8
Riesgo en Desarrollo: Baja	Puntos Reales: 0.7
Descripción: Este requisito se produce cuando un usuario autenticado con los permisos que se requieren desea revisar el perfil de tesis.	

Tareas de Ingeniería (R4)

Tarea de Ingeniería	
Número Tarea: 1	Número Historia de Usuario: HU-4
Nombre de la Tarea: Crear una vista para mostrar los perfiles sin revisar.	
Tipo de Tarea: Programación	Puntos estimados: 0.8
Fecha Inicio: 28/1/2011	Fecha Fin: 29/1/2011
Programador Responsable: Leodan Cabrera y Leticia Rodríguez	
Descripción: Se crea una vista donde se le muestran al revisor todos los perfiles que están sin revisar.	

Tarea de Ingeniería	
Número Tarea: 2	Número Historia de Usuario: HU-4
Nombre de la Tarea: Instalar y configurar el módulo workflow.	
Tipo de Tarea: Programación	Puntos estimados: 0.8
Fecha Inicio: 29/1/2011	Fecha Fin: 29/1/2011
Programador Responsable: Leodan Cabrera y Leticia Rodríguez	

PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA

Descripción: Se instala el módulo workflow para crear el flujo de trabajo sobre el tipo de contenido perfil.

R5: Mostrar cortes realizados

Historia de Usuario	
Número: HU-5	Nombre Historia de Usuario: Mostrar cortes realizados
Modificación de Historia de Usuario Número: 1	
Usuario: Leodan Cabrera Rodríguez	Iteración Asignada: 2
Prioridad en Negocio: Alta	Puntos Estimados: 0.6
Riesgo en Desarrollo: Bajo	Puntos Reales: 0.6
Descripción: Este requisito se produce cuando un usuario autenticado desea consultar el resultado de la revisión del corte de tesis.	
Observaciones:	

Tareas de Ingeniería (R5)

Tarea de Ingeniería	
Número Tarea: 1	Número Historia de Usuario: HU-5
Nombre de la Tarea: Crear una vista para mostrar las actas realizadas en los cortes.	
Tipo de Tarea: Programación	Puntos estimados: 0.6
Fecha Inicio: 2/2/2011	Fecha Fin: 2/2/2011
Programador Responsable: Leodan Cabrera y Leticia Rodríguez	
Descripción: Se crea una vista donde se muestran todas las actas de los cortes.	

R6: Asignar perfil de tesis a tesista.

Historia de Usuario	
Número: HU-6	Nombre Historia de Usuario: Asignar perfil de tesis a tesista.
Modificación de Historia de Usuario Número: 1	
Usuario: Leodan Cabrera Rodríguez	Iteración Asignada: 1

PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA

Prioridad en Negocio: Muy Alta	Puntos Estimados: 0.2
Riesgo en Desarrollo: Bajo	Puntos Reales: 0.2
Descripción: Este requisito se produce cuando un usuario autenticado con determinados permisos procede a vincular un perfil de tesis a un tesista.	
Observaciones: El estudiante interesado en el perfil se presenta ante el profesor que creó el perfil y este es el encargado de asignarlo.	

Tareas de Ingeniería (R6)

Tarea de Ingeniería	
Número Tarea: 1	Número Historia de Usuario: HU-6
Nombre de la Tarea: Crear una vista para mostrar los perfiles aprobados.	
Tipo de Tarea: Programación	Puntos estimados: 0.2
Fecha Inicio: 2/2/2011	Fecha Fin: 3/2/2011
Programador Responsable: Leodan Cabrera y Leticia Rodríguez	
Descripción: Se crea una vista para facilitar la asignación del perfil de tesis al estudiante.	

R7: Gestionar tribunal de tesis

Historia de Usuario	
Número: HU-7	Nombre Historia de Usuario: Gestionar tribunal de tesis
Modificación de Historia de Usuario Número: 1	
Usuario: Leodan Cabrera Rodríguez	Iteración Asignada: 2
Prioridad en Negocio: Alta	Puntos Estimados: 0.5
Riesgo en Desarrollo: Baja	Puntos Reales: 0.4
Descripción: Este requisito se produce cuando un usuario autenticado con determinados permisos procede a conformar tribunales de tesis.	
Observaciones: Un tribunal es creado por un jefe de Departamento.	

PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA

Tareas de Ingeniería (R7)

Tarea de Ingeniería	
Número Tarea: 1	Número Historia de Usuario: HU-7
Nombre de la Tarea: Crear el tipo de contenido tribunal.	
Tipo de Tarea: Programación	Puntos estimados: 0.3
Fecha Inicio: 4/2/2011	Fecha Fin: 4/2/2011
Programador Responsable: Leodan Cabrera y Leticia Rodríguez	
Descripción: Se crea el tipo de contenido tribunal mediante la utilización de módulo CCK.	

Tarea de Ingeniería	
Número Tarea: 2	Número Historia de Usuario: HU-7
Nombre de la Tarea: Crear una vista para buscar perfiles.	
Tipo de Tarea: Programación	Puntos estimados: 0.1
Fecha Inicio: 4/2/2011	Fecha Fin: 5/2/2011
Programador Responsable: Leodan Cabrera y Leticia Rodríguez	
Descripción: Se crea una vista para buscar los perfiles que están aprobados y que no han sido asignados a otro tribunal.	

Tarea de Ingeniería	
Número Tarea: 3	Número Historia de Usuario: HU-7
Nombre de la Tarea: Instalar y configurar el módulo multiselect.	
Tipo de Tarea: Programación	Puntos estimados: 0.1
Fecha Inicio: 5/2/2011	Fecha Fin: 5/2/2011
Programador Responsable: Leodan Cabrera y Leticia Rodríguez	
Descripción: Se instala y configura el módulo multiselect que permite seleccionar múltiples elementos a la vez.	

PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA

R8: Asignar perfiles a tribunal de tesis.

Historia de Usuario	
Número: HU-8	Nombre Historia de Usuario: Asignar perfiles a tribunal de tesis.
Modificación de Historia de Usuario Número: 1	
Usuario: Leodan Cabrera Rodríguez y Leticia Rodríguez Viera	Iteración Asignada: 2
Prioridad en Negocio: Alta	Puntos Estimados: 0.8
Riesgo en Desarrollo: Bajo	Puntos Reales: 0.8
Descripción: Este requisito se produce cuando un usuario autenticado con determinados permiso procede a vincular tribunales de tesis con perfiles.	
Observaciones: Un perfil es asignado a un tribunal por un jefe de Departamento.	

Tareas de Ingeniería (R8)

Tarea de Ingeniería	
Número Tarea: 1	Número Historia de Usuario: HU-8
Nombre de la Tarea: Crear una vista para buscar tribunales.	
Tipo de Tarea: Programación	Puntos estimados: 0.3
Fecha Inicio: 6/2/2011	Fecha Fin: 7/2/2011
Programador Responsable: Leodan Cabrera y Leticia Rodríguez	
Descripción: Se crea una vista donde se muestran todos los tribunales creados que no tienen tesis asignadas.	

Tarea de Ingeniería	
Número Tarea: 2	Número Historia de Usuario: HU-8
Nombre de la Tarea: Crear una vista para buscar perfiles.	
Tipo de Tarea: Programación	Puntos estimados: 0.2
Fecha Inicio: 7/2/2011	Fecha Fin: 7/2/2011

PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA

Programador Responsable: Leodan Cabrera y Leticia Rodríguez

Descripción: Se crea una vista para buscar los perfiles que están aprobados y que no han sido asignados a otro tribunal.

R9: Crear acta de evaluación.

Historia de Usuario	
Número: HU-9	Nombre Historia de Usuario: Crear acta de evaluación.
Modificación de Historia de Usuario Número: 1	
Usuario: Leodan Cabrera Rodríguez y Leticia Rodríguez Viera	Iteración Asignada: 2
Prioridad en Negocio: Alta	Puntos Estimados: 1
Riesgo en Desarrollo: Bajo	Puntos Reales: 1
Descripción: Este requisito se produce cuando un usuario autenticado relacionado como miembro de tribunal de tesis a alguna tesis, proceda a registrar, obtener o modificar el acta del corte de tesis.	
Observaciones:	

Tareas de Ingeniería (R9)

Tarea de Ingeniería	
Número Tarea: 1	Número Historia de Usuario: HU-9
Nombre de la Tarea: Crear e instalar el módulo crear_acta.	
Tipo de Tarea: Programación	Puntos estimados: 0.4
Fecha Inicio: 10/2/2011	Fecha Fin: 10/2/2011
Programador Responsable: Leodan Cabrera y Leticia Rodríguez	
Descripción: Se crea y configura el módulo crear_acta para gestionar los datos de las actas del corte de tesis.	

PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA

R10: Gestionar cronograma.

Historia de Usuario	
Número: HU-10	Nombre Historia de Usuario: Gestionar cronograma.
Modificación de Historia de Usuario Número: 1	
Usuario: Leodan Cabrera y Leticia Rodríguez	Iteración Asignada: 3
Prioridad en Negocio: Media	Puntos Estimados: 0.6
Riesgo en Desarrollo: Bajo	Puntos Reales: 0.6
Descripción: Este requisito se produce cuando un usuario autenticado con permisos procede a planificar una evaluación (corte de tesis, pre defensa o defensa) de perfiles de tesis.	
Observaciones: El cronograma es creado por el jefe de centro.	

Tareas de Ingeniería (R10)

Tarea de Ingeniería	
Número Tarea: 1	Número Historia de Usuario: HU-10
Nombre de la Tarea: Crear el tipo de contenido cronograma.	
Tipo de Tarea: Programación	Puntos estimados: 0.3
Fecha Inicio: 10/3/2011	Fecha Fin: 10/3/2011
Programador Responsable: Leodan Cabrera y Leticia Rodríguez	
Descripción: Se crea el tipo de contenido cronograma mediante el módulo CCK.	

Tarea de Ingeniería	
Número Tarea: 2	Número Historia de Usuario: HU-10
Nombre de la Tarea: Crear una vista para mostrar el cronograma.	
Tipo de Tarea: Programación	Puntos estimados: 0.3
Fecha Inicio: 11/3/2011	Fecha Fin: 11/3/2011

PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA

Programador Responsable: Leodan Cabrera y Leticia Rodríguez

Descripción: Se crea una vista donde se muestra los datos del cronograma creado.

R11: Mostrar banco de perfiles reducidos

Historia de Usuario	
Número: HU-11	Nombre Historia de Usuario: Mostrar banco de perfiles reducidos
Modificación de Historia de Usuario Número: 1	
Usuario: Leodan Cabrera y Leticia Rodríguez	Iteración Asignada: 1
Prioridad en Negocio: Muy Alta	Puntos Estimados: 0.3
Riesgo en Desarrollo: Bajo	Puntos Reales: 0.2
Descripción: Este requisito se produce cuando un usuario autenticado como estudiante militar desea consultar un perfil reducido.	
Observaciones: Solo se les mostrará a los estudiantes del MINIT y la FAR.	

Tareas de Ingeniería (R11)

Tarea de Ingeniería	
Número Tarea: 1	Número Historia de Usuario: HU-11
Nombre de la Tarea: Crear una vista para mostrar el banco de perfiles reducido.	
Tipo de Tarea: Programación	Puntos estimados:
Fecha Inicio: 25/2/2011	Fecha Fin: 25/2/2011
Programador Responsable: Leodan Cabrera y Leticia Rodríguez	
Descripción: A través de una vista se muestra el banco de perfiles reduciendo.	

R12: Gestionar centro.

Historia de Usuario	
Número: HU-12	Nombre Historia de Usuario: Gestionar centro
Modificación de Historia de Usuario Número: Ninguna	

PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA

Usuario: Leodan Cabrera y Leticia Rodríguez	Iteración Asignada: 2
Prioridad en Negocio: Alta	Puntos Estimados: 0.5
Riesgo en Desarrollo: Bajo	Puntos Reales: 0.3
Descripción: Este requisito se produce cuando un usuario autenticado con los permisos que requiere desea crear, obtener, modificar y eliminar un centro.	
Observaciones:	

Tareas de Ingeniería (R12)

Tarea de Ingeniería	
Número Tarea: 1	Número Historia de Usuario: HU-12
Nombre de la Tarea: Crear el tipo de contenido Centro.	
Tipo de Tarea: Programación	Puntos estimados:
Fecha Inicio: 27/2/2011	Fecha Fin: 27/2/2011
Programador Responsable: Leodan Cabrera y Leticia Rodríguez	
Descripción: Se crea el tipo de contenido Centro.	

R13: Gestionar departamento.

Historia de Usuario	
Número: HU-13	Nombre Historia de Usuario: Gestionar departamento
Modificación de Historia de Usuario Número: Ninguna	
Usuario: Leodan Cabrera y Leticia Rodríguez	Iteración Asignada: 2
Prioridad en Negocio: Alta	Puntos Estimados: 0.5
Riesgo en Desarrollo: Bajo	Puntos Reales: 0.3
Descripción: Este requisito se produce cuando un usuario autenticado con los permisos que requiere desea crear, obtener, modificar y eliminar un Departamento.	
Observaciones:	

PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA

Tareas de Ingeniería (R13)

Tarea de Ingeniería	
Número Tarea: 1	Número Historia de Usuario: HU-13
Nombre de la Tarea: Crear el tipo de contenido Departamento.	
Tipo de Tarea: Programación	Puntos estimados:
Fecha Inicio: 10/3/2011	Fecha Fin: 10/3/2011
Programador Responsable: Leodan Cabrera y Leticia Rodríguez	
Descripción: Se crea el tipo de contenido Departamento.	

R14 Gestionar facultad.

Historia de Usuario	
Número: HU-14	Nombre Historia de Usuario: Gestionar facultad
Modificación de Historia de Usuario Número: Ninguna	
Usuario: Leodan Cabrera y Leticia Rodríguez	Iteración Asignada: 2
Prioridad en Negocio: Alta	Puntos Estimados: 0.4
Riesgo en Desarrollo: Bajo	Puntos Reales: 0.4
Descripción: Este requisito se produce cuando un usuario autenticado con los permisos que requiere desea crear, obtener, modificar y eliminar una Facultad.	
Observaciones:	

Tareas de Ingeniería (R14)

Tarea de Ingeniería	
Número Tarea: 1	Número Historia de Usuario: HU-14
Nombre de la Tarea: Crear el tipo de contenido Facultad.	
Tipo de Tarea: Programación	Puntos estimados: 0.4
Fecha Inicio: 16/3/2011	Fecha Fin: 16/3/2011
Programador Responsable: Leodan Cabrera y Leticia Rodríguez	

PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA

Descripción: Se crea el tipo de contenido Facultad.

R15 Gestionar ddatos del pprofesor.

Historia de Usuario	
Número: HU-15	Nombre Historia de Usuario: Gestionar datos del profesor.
Modificación de Historia de Usuario Número: Ninguna	
Usuario: Leodan Cabrera y Leticia Rodríguez	Iteración Asignada: 2
Prioridad en Negocio: Alta	Puntos Estimados: 0.3
Riesgo en Desarrollo: Bajo	Puntos Reales: 0.3
Descripción: Este requisito se produce cuando un usuario autenticado con los permisos que requiere desea crear, obtener, modificar y eliminar los datos de un profesor.	
Observaciones:	

Tareas de Ingeniería (R15)

Tarea de Ingeniería	
Número Tarea: 1	Número Historia de Usuario: HU-15
Nombre de la Tarea: Crear el tipo de contenido datos del profesor.	
Tipo de Tarea: Programación	Puntos estimados:
Fecha Inicio: 17/3/2011	Fecha Fin: 17/3/2011
Programador Responsable: Leodan Cabrera y Leticia Rodríguez	
Descripción: Se crea el tipo de contenido datos del profesor.	

2.9 Plan de release.

Release	Descripción de la iteración	Orden de la HU a implementar	Duración total
---------	-----------------------------	------------------------------	----------------

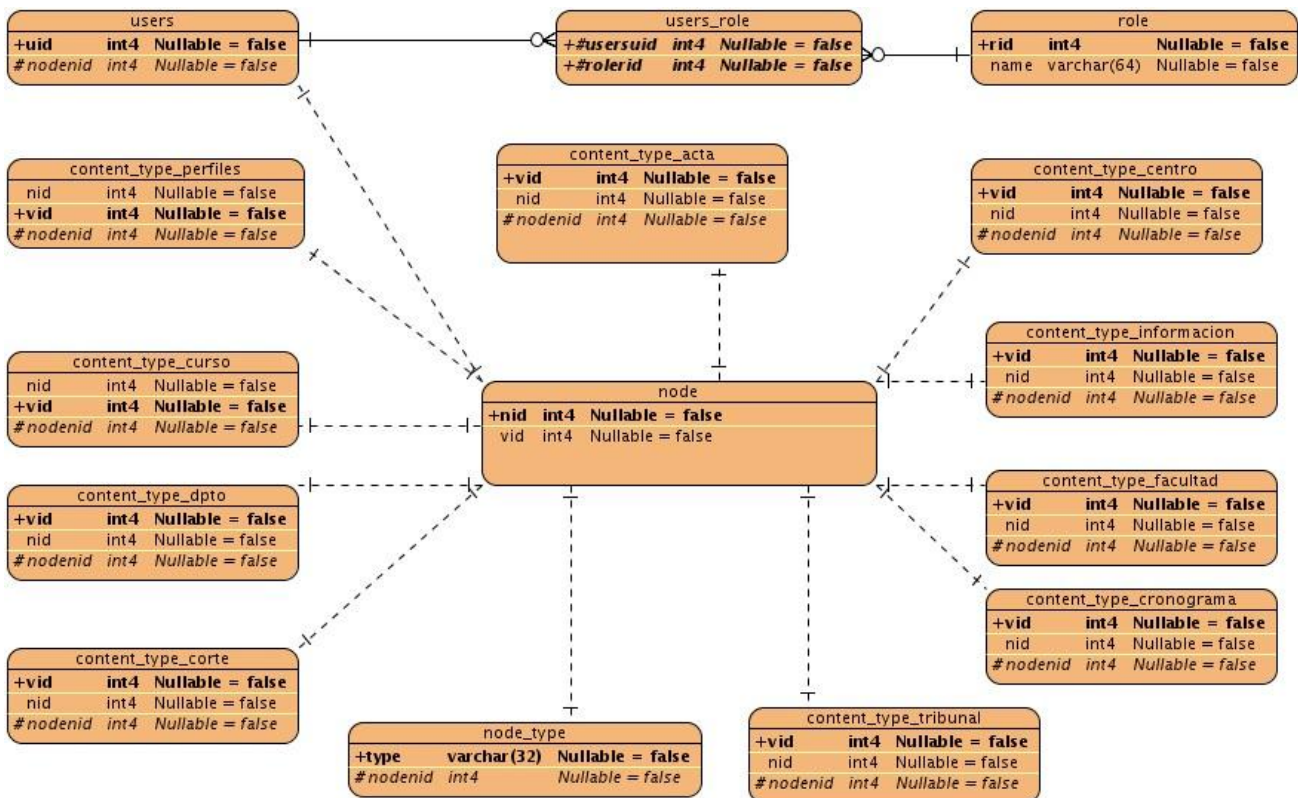
PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA

Iteración 1	En esta iteración se implementaran las HU 1, 2,3 ,6 ,11 lo cual permite dar comienzo al desarrollo del sitio.	HU-1, HU-2, HU-3, HU-6, HU-11.	3
Iteración 2	En esta iteración durante la implementación del sitio, se dará cumplimiento a las HU 4, 5, 7, 8, 9, 12, 13, 14, 15 y esto se integrará con lo desarrollado anteriormente.	HU-4, HU-5, HU-7, HU-8, HU-9, HU-12, HU-13, HU-14, HU-15.	8
Iteración 3	En esta iteración se implementará la HU 10 y se determinará la eficiencia de la solución complementando todo lo desarrollado anteriormente. Luego se realizaran las pruebas a la aplicación.	HU-10	3

2.10 Modelo de Datos.

El modelo de datos es uno de los elementos más importantes a la hora de iniciar el desarrollo de cualquier proyecto. Esta es la estructura, sobre la que realmente reside la verdadera esencia de la aplicación. Incluso determina si esta va a cumplir con su verdadero objetivo. El modelado de datos es una técnica independiente de la implementación a la base de datos. Esto es importante, ya que siempre busca que se saque el máximo provecho de diversas herramientas.

PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA



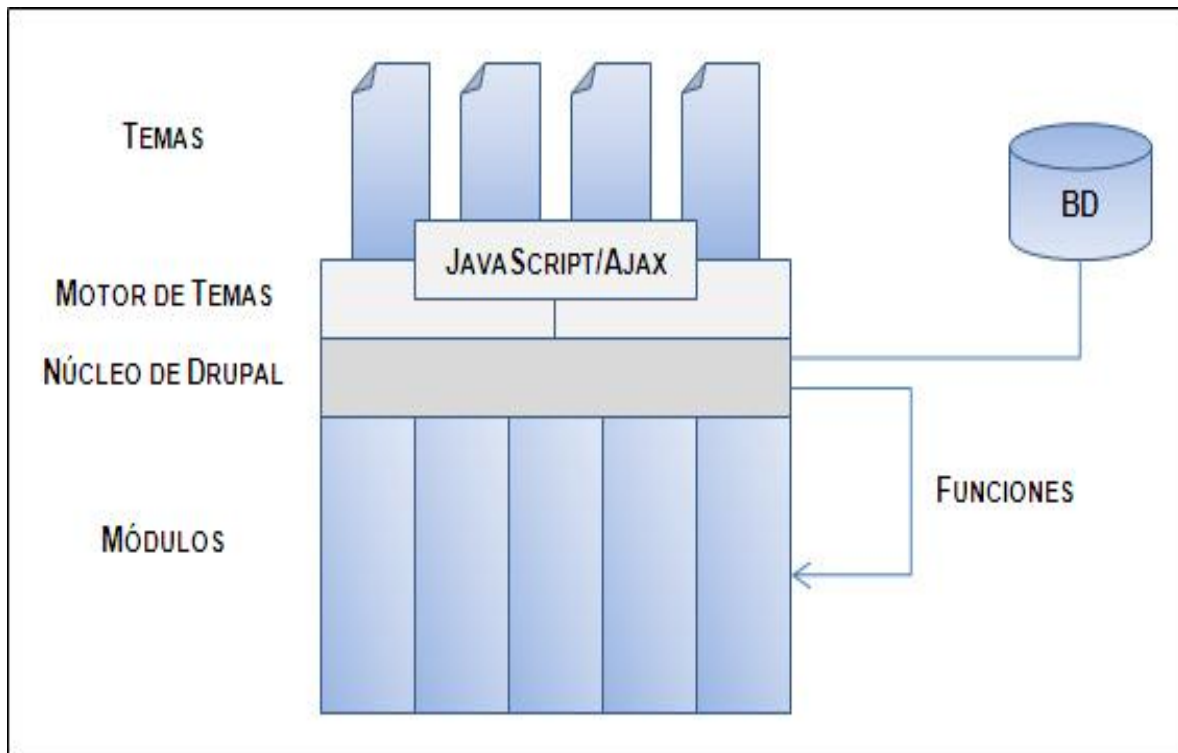
2.11 Patrones y arquitectura

Un patrón es un conjunto de información que aporta la solución a un problema que se presenta en un contexto determinado. Considerando que Drupal fue el CMS usado en la implementación del sistema, los patrones y arquitectura se heredarán del mismo y como este utiliza Modelo Vista Controlador (MVC), este será el patrón utilizado en la aplicación.

La arquitectura MVC separa la lógica de negocio (el modelo) y la presentación (la vista) por lo que se consigue un mantenimiento más sencillo de las aplicaciones. El controlador se encarga de aislar al modelo y a la vista de los detalles del protocolo utilizado para las peticiones. El modelo se encarga de la abstracción de la lógica relacionada con los datos, haciendo que la vista y las acciones sean independientes de, por ejemplo, el tipo de gestor de bases de datos utilizado por la aplicación.

- **Modelo:** Esta es la representación específica de la información con la cual el sistema opera.
- **Vista:** Este presenta el modelo en un formato adecuado para interactuar, usualmente la interfaz del usuario.
- **Controlador:** Este responde a eventos, usualmente acciones del usuario, e invoca peticiones al modelo y, probablemente, a la vista.

PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA



En la figura anterior se muestra como es la estructura en capas del CMS Drupal donde las vista está formada por los por los Temas, Motor de Temas y Java Script/Ajax; el modelo lo conforma la base de datos y el controlador está formado por el Núcleo de Drupal y los Módulos. Las Funciones están para facilitar la comunicación entre el Núcleo de Drupal y los Módulos.

Los patrones de diseño son la base de las soluciones a problemas comunes en el desarrollo de software. Drupal utiliza algunos patrones Gang of Four (GoF) con el objetivo de que el diseño y la aplicación sean más robustos. Los mismo se clasifican en:

- Creacionles: Solucionan problemas de crear instancias.
- Estructurales: Solucionan problemas de composición de clases y objetos.
- Comportamiento: Solucionan problemas respecto a la interacción y responsabilidades entre clases y objetos.

Los patrones que el CMS Drupal pone de manifiesto en su funcionamiento son:

- Instancia Única: Se utiliza para garantizar que una clase sólo tenga una única instancia y para facilitar un punto de acceso global a la misma.

PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA

- Decorador: Responde a la necesidad de añadir dinámicamente nuevas funcionalidades a una clase. Esto nos permite no tener que crear sucesivas clases que hereden de la primera incorporando la nueva funcionalidad, sino otras que la implementan y se asocian a la primera.
- Puente: El patrón es utilizado para desacoplar una abstracción de su implementación, de manera que ambas puedan ser modificadas independientemente sin necesidad de alterar por ello la otra.
- Observador: Define una dependencia entre objetos, de forma tal que cuando uno de los objetos cambie su estado el observador notifica este cambio a todos los otros dependientes. El objetivo principal es el de desacoplar la clase de los objetos clientes del objeto aumentando la modularidad del lenguaje.

2.12 Seguridad en Drupal

Drupal proporciona una capa de abstracción de bases de datos delgada para proporcionar a los desarrolladores la capacidad de soportar múltiples servidores de bases de datos fácilmente. La intención de esta capa es la de preservar la sintaxis y el poder de SQL tanto como sea posible, dejando el control de Drupal a las piezas de las consultas que deben ser escritos de manera diferente para diferentes servidores y proporcionar los controles básicos de seguridad mediante la utilización de las funciones de la interfaz de programación de aplicaciones de Drupal (API de Drupal). Se utilizan unos determinados modificadores (%d enteros, %s cadenas, etc) para validar los argumentos que se le pasan y evitar las inyecciones SQL. Tras la consulta, se le pasan los argumentos por orden de aparición y Drupal es el responsable de comprobar los valores que reciben los modificadores, retirando las comillas y testeando los valores reales con los que debería tener.

```
$result = db_query ("SELECT title FROM {node} WHERE type='%s' AND uid=%d", $tipo, $user->uid);
```

Drupal es un CMS multiusuario, el cual maneja los permisos de los usuarios en dependencia del rol que este posea, donde un mismo usuario puede tener uno o varios roles, teniendo acceso entonces a todas las acciones que se le definan a los roles correspondientes. Para el manejo de los permisos, Drupal se auxilia del módulo Content Permissions, del paquete CCK, con el cual se le da los permisos a los diferentes roles existentes en el sistema en dependencia de las acciones que cada uno pueda llevar a cabo. Este módulo permite dar permiso a cada campo y a cada tipo de contenido individualmente que se haya creado en el sistema. Esto posibilita que los usuarios solo puedan hacer y acceder a las acciones que se le definan al rol que poseen.

2.13 Conclusiones Parciales

En este capítulo se analizaron las características principales para la gestión de las tesis que actualmente se llevan a cabo en la facultad, a partir de la comprensión de los procesos de dominio, se definieron las principales funcionalidades que debe tener el sistema a desarrollar. En el mismo se provee de una visión general de lo que el sistema debe hacer, por lo que de esta forma quedan creadas las condiciones para comenzar la implementación del sistema.

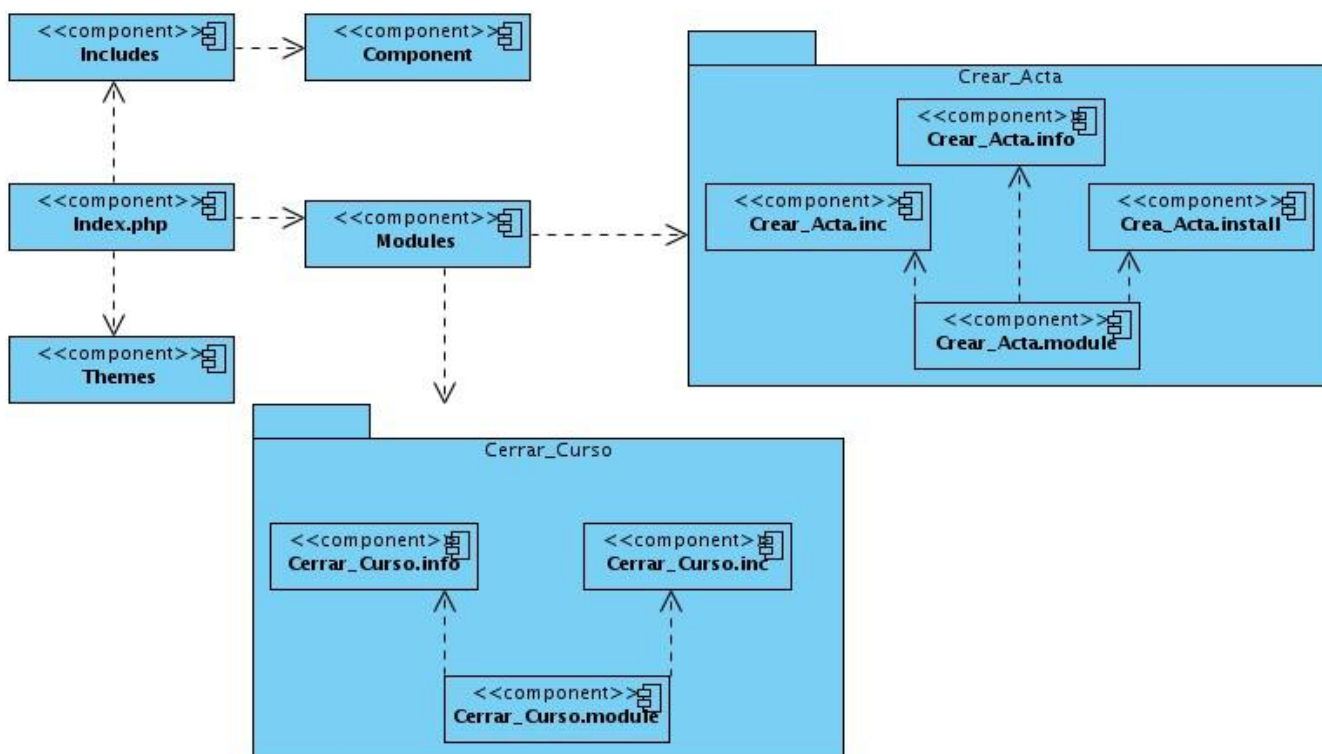
CAPÍTULO 3: CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA

3.1 Introducción

El presente capítulo está enfocado en la etapa de trabajo de construcción para dar solución a los requisitos especificados. Se describen los elementos necesarios para la implementación, partiendo del resultado obtenido del diseño. Además serán expuestos diferentes artefactos como son: Diagramas de componentes y Diagrama de despliegue proporcionando una visión de cómo quedará distribuida la aplicación. Se muestra el modelo de prueba donde se hace la descripción de los casos de prueba por cada caso de uso.

3.2 Diagramas de Componentes

En los diagramas de componentes se representan un conjunto de elementos del modelo de implementación tales como componentes y sus relaciones. Este representa cómo un sistema es dividido en componentes mostrando las organizaciones y dependencias lógicas entre componentes de software, sean estos componentes de código fuente, binario o ejecutables. Se utilizan para modelar la vista estática y documentar la arquitectura de sistema.



3.3 Diagrama de Paquetes

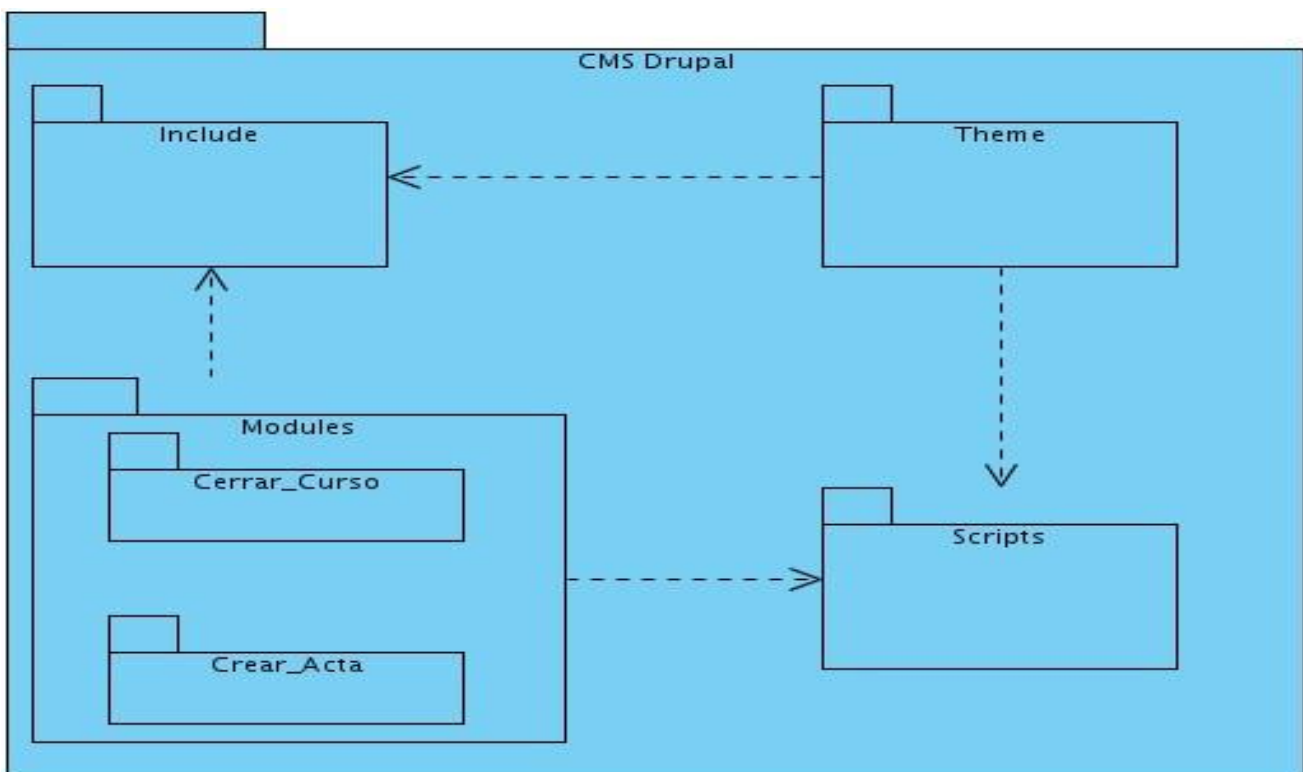
El diagrama que se muestra a continuación es el diagrama de paquetes que genera el CMS Drupal, para mejor comprensión se decidió describir por paquetes los principales directorios incluidos en el core de este CMS.

Include: Contiene un conjunto de ficheros imprescindibles para el correcto funcionamiento, como el fichero DataBase, que provee las funcionalidades de acceso a la base de datos del CMS Drupal.

Themes: Contiene las plantillas de diseño, si se desea incluir alguna plantilla personalizada u otra de las que se encuentran en el sitio de Drupal, solo se necesita copiar dicha plantilla en esa carpeta.

Scripts: Contiene un conjunto de ficheros necesarios orientados al aspecto visual, CSS y JavaScript para el correcto funcionamiento de la aplicación.

Modules: Comprende todos los módulos del CMS Drupal que permiten todas sus funcionalidades, si se desea extender las funcionalidades que posee el mismo, solo se necesita colocar en esta carpeta el módulo correspondiente un vez esté implementado.



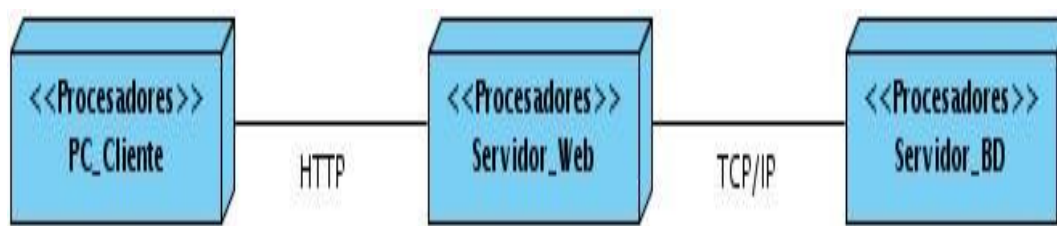
3.4 Diagrama de Despliegue

En el diagrama de despliegue que aparece a continuación se muestran las relaciones físicas de los distintos nodos que componen el sistema y el reparto de los componentes sobre dichos nodos. Un nodo es un recurso de ejecución tal como un computador, un dispositivo o memoria. Los estereotipos permiten precisar la naturaleza del equipo:

- Dispositivos
- Procesadores
- Memoria

A continuación se explican los recursos presentes en cada nodo del diagrama:

- Nodo PC Cliente a través del cual los clientes se conectarán al sistema y podrán operar en el mismo. Esta PC deberá contar con el sistema operativo Windows XP o GNU/Linux además de los navegadores Internet Explorer o Mozilla Firefox que le permitirán al cliente tener acceso a la aplicación.
- Nodo Servidor Web, el mismo deberá contar con el software base GNU/Linux Distribución Debian 4 Etch o superior o Ubuntu 8.04 o superior además como servidor Web Apache2.2 y el lenguaje de programación PHP5.
- Nodo Servidor Base de Datos este servidor deberá utilizar como software base GNU/Linux Distribución Debian 4 Etch o Ubuntu 8.04 y como Sistema Gestor de Base de Datos PostgreSQL 8.3.



3.5 Modelos de prueba

El objetivo de las pruebas de aceptación es validar que un sistema cumple con el funcionamiento esperado y permitir al usuario de dicho sistema que determine su aceptación, desde el punto de vista de su funcionalidad y rendimiento. Las pruebas de aceptación son definidas por el usuario del sistema y preparadas por el equipo de desarrollo, aunque la ejecución y aprobación final corresponden al usuario. La validación del sistema se consigue mediante la realización de pruebas de caja negra que demuestran

CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA

la conformidad con los requisitos y que se recogen en el plan de pruebas, el cual define las verificaciones a realizar y los casos de prueba asociados. Dicho plan está diseñado para asegurar que se satisfacen todos los requisitos funcionales especificados por el usuario teniendo en cuenta también los requisitos no funcionales relacionados con el rendimiento, seguridad de acceso al sistema, a los datos y procesos, así como a los distintos recursos del sistema.

Nombre del caso de uso: Autenticar usuario.

Caso de Prueba de Aceptación	
Código Caso de Prueba: HU-1_P1	Nombre Historia de Usuario: Autenticar usuario.
Nombre de la persona que realiza la prueba: Leodan Cabrera Rodríguez	
Descripción de la Prueba: Prueba de autenticación de usuarios.	
Condiciones de Ejecución: Los usuarios deben tener un usuario y una contraseña para acceder al sistema.	
Entrada / Pasos de ejecución: El usuario procede a loguearse en el sitio con su usuario y contraseña. 1-Dar clic en el botón y dejar campos vacíos. 2-Dar clic en el botón y escribir mal usuario o contraseña. 3-Dar clic en el botón y tener los datos correctos.	
Resultado Esperado: 1-El sistema muestra un cartel: No puede dejar campos vacíos. 2-El sistema muestra un cartel: Usuario o contraseña incorrecta. 3-El usuario puede acceder al sistema.	
Evaluación de la Prueba: Prueba satisfactoria.	

Nombre del caso de uso: Crear Perfil de Tesis.

Caso de Prueba de Aceptación	
Código Caso de Prueba: HU-2_P2	Nombre Historia de Usuario: Gestionar Perfiles.

CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA

Nombre de la persona que realiza la prueba: Leodan Cabrera Rodríguez
Descripción de la Prueba: Prueba a la funcionalidad Crear Perfiles de Tesis.
Condiciones de Ejecución: El usuario tiene que estar autenticado en el sitio y con los permisos correspondientes para realizar esta acción.
Entrada / Pasos de ejecución: Una vez el usuario registrado en el sitio y con los permisos de acceder a las interfaces correspondientes, este puede escoger que acción realizar. El usuario escoge la opción de crear un nuevo perfil. 1-Dar clic en el botón y dejar campos obligatorios en blanco. 2-Dar clic en el botón con todos los datos correctos.
Resultado Esperado: 1-El sistema muestra un cartel: No puede dejar campos en blanco. 2-Con todos los datos correctos se crea un nuevo perfil de tesis que se guarda en la base de datos.
Evaluación de la Prueba: Prueba satisfactoria.

Nombre del caso de uso: Eliminar Perfil de Tesis.

Caso de Prueba de Aceptación	
Código Caso de Prueba: HU-2_P3	Nombre Historia de Usuario: Gestionar Perfiles.
Nombre de la persona que realiza la prueba: Leodan Cabrera Rodríguez	
Descripción de la Prueba: Prueba a la funcionalidad Eliminar Perfiles de Tesis.	
Condiciones de Ejecución: El usuario tiene que estar autenticado en el sitio y con los permisos correspondientes para realizar esta acción.	
Entrada / Pasos de ejecución: Una vez el usuario registrado en el sitio y con los permisos de acceder a las interfaces correspondientes, este puede escoger que acción realizar. El usuario escoge la opción de eliminar uno de los perfiles que ha creado, lo busca y lo elimina.	
Resultado Esperado: Una vez eliminado el perfil escogido, este desaparece de los perfiles del profesor y se actualiza la base de datos.	

CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA

Evaluación de la Prueba: Prueba satisfactoria.

Nombre del caso de uso: Actualizar Perfil de Tesis.

Caso de Prueba de Aceptación

Código Caso de Prueba: HU-2_P4 **Nombre Historia de Usuario:** Gestionar Perfiles.

Nombre de la persona que realiza la prueba: Leodan Cabrera Rodríguez

Descripción de la Prueba: Prueba a la funcionalidad Actualizar Perfiles de Tesis.

Condiciones de Ejecución: El usuario tiene que estar autenticado en el sitio y con los permisos correspondientes para realizar esta acción.

Entrada / Pasos de ejecución: Una vez el usuario registrado en el sitio y con los permisos de acceder a las interfaces correspondientes, este puede escoger que acción realizar. Si el usuario escoge la opción de actualizar uno de sus perfiles, se le muestra una interfaz con todos sus perfiles, escoge el que desea modificar y edita sus campos.

Resultado Esperado: Una vez editado el perfil seleccionado, los cambios se guardan en la base de datos.

Evaluación de la Prueba: Prueba satisfactoria.

Nombre del caso de uso: Consultar Perfil de Tesis.

Caso de Prueba de Aceptación

Código Caso de Prueba: HU-2_P5 **Nombre Historia de Usuario:** Gestionar Perfiles.

Nombre de la persona que realiza la prueba: Leodan Cabrera Rodríguez

Descripción de la Prueba: Prueba a la funcionalidad Consultar Perfiles de Tesis.

Condiciones de Ejecución: El usuario tiene que estar autenticado en el sitio y con los permisos correspondientes para realizar esta acción.

Entrada / Pasos de ejecución: Una vez el usuario registrado en el sitio y con los permisos de acceder a las interfaces correspondientes, este puede escoger que acción realizar. Si el usuario escoge la opción de consultar sus perfiles, se le muestra una interfaz con todos sus perfiles.

CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA

Resultado Esperado: Se le muestran al usuario todos sus perfiles.
Evaluación de la Prueba: Prueba satisfactoria.

Nombre del caso de uso: Mostrar banco de perfiles.

Caso de Prueba de Aceptación	
Código Caso de Prueba: HU-3_P6	Nombre Historia de Usuario: Mostrar banco de perfiles.
Nombre de la persona que realiza la prueba: Leodan Cabrera Rodríguez	
Descripción de la Prueba: Prueba a la funcionalidad de Mostrar Perfiles Disponibles.	
Condiciones de Ejecución: El usuario con el rol de estudiante tiene que estar logueado en el sistema.	
Entrada / Pasos de ejecución: Una vez el usuario registrado en el sitio y con los permisos de acceder a las interfaces correspondientes, este puede escoger que acción realizar. Si el usuario escoge la opción de Mostrar Perfiles, se le muestra una interfaz con todos los perfiles disponibles.	
Resultado Esperado: Se le muestran al usuario todos los perfiles que están disponibles en el sistema.	
Evaluación de la Prueba: Prueba satisfactoria.	

Nombre del caso de uso: Revisar perfil.

Caso de Prueba de Aceptación	
Código Caso de Prueba: HU-4_P7	Nombre Historia de Usuario: Revisar perfil.
Nombre de la persona que realiza la prueba: Leodan Cabrera Rodríguez	
Descripción de la Prueba: Prueba a la funcionalidad Revisar Perfil.	
Condiciones de Ejecución: El usuario tiene que estar loqueado en el sistema y tener el rol de revisor.	

CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA

Entrada / Pasos de ejecución: Una vez el usuario registrado en el sitio y con los permisos de acceder a las interfaces correspondientes, este puede escoger que acción realizar. Si el usuario escoge la opción de Revisar Perfil, se le muestra una interfaz donde se encuentran todos los perfiles que debe revisar.
Resultado Esperado: El perfil es revisado y puede ser aceptado o rechazado, todos estos cambios se registran en la base de datos.
Evaluación de la Prueba: Prueba satisfactoria.

Nombre del caso de uso: Mostrar cortes realizados

Caso de Prueba de Aceptación	
Código Caso de Prueba: HU-5_P8	Nombre Historia de Usuario: Mostrar cortes realizados.
Nombre de la persona que realiza la prueba: Leodan Cabrera Rodríguez	
Descripción de la Prueba: Prueba a la funcionalidad de Mostrar Cortes Realizados.	
Condiciones de Ejecución: El usuario tiene que estar registrado en el sistema y tener el rol de profesor capacitado.	
Entrada / Pasos de ejecución: Una vez el usuario registrado en el sitio y con los permisos de acceder a las interfaces correspondientes, este puede escoger que acción realizar. Si escoge la opción de Mostrar Cortes Realizados, se le muestra una interfaz donde se encuentran todos los cortes que ha realizado.	
Resultado Esperado: Se le muestran al usuario todos los cortes que ha realizado.	
Evaluación de la Prueba: Prueba satisfactoria.	

Nombre del caso de uso: Crear Tribunal de Tesis

Caso de Prueba de Aceptación	
Código Caso de Prueba: HU-7_P9	Nombre Historia de Usuario: Gestionar tribunal de tesis.
Nombre de la persona que realiza la prueba: Leodan Cabrera Rodríguez	
Descripción de la Prueba: Prueba a la funcionalidad Crear Tribunal de Tesis.	
Condiciones de Ejecución: El usuario tiene que estar registrado en el sitio y con los permisos	

CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA

correspondientes.
Entrada / Pasos de ejecución: Una vez el usuario registrado en el sitio y con los permisos de acceder a las interfaces correspondientes, este puede escoger que acción realizar. El usuario escoge la opción de crear un nuevo Tribunal de Tesis. 1-El usuario da clic en el botón y deja campos en blancos que son obligatorios. 2-El usuario da clic en el botón con todos los campos correctos.
Resultado Esperado: 1-El sistema muestra un cartel: No puede dejar campos en blanco. 2-Se crea el nuevo tribunal almacenándose en base de datos.
Evaluación de la Prueba: Prueba satisfactoria.

Nombre del caso de uso: Eliminar Tribunal de Tesis

Caso de Prueba de Aceptación	
Código Caso de Prueba: HU-7_P10	Nombre Historia de Usuario: Gestionar Tribunal de Tesis.
Nombre de la persona que realiza la prueba: Leodan Cabrera Rodríguez	
Descripción de la Prueba: Prueba a la funcionalidad de Eliminar Perfil de Tesis.	
Condiciones de Ejecución: El usuario tiene que estar loqueado y con los permisos correspondientes.	
Entrada / Pasos de ejecución: Una vez el usuario registrado en el sitio y con los permisos de acceder a las interfaces correspondientes, este puede escoger que acción realizar. Si el usuario escoge la opción de Eliminar un Tribunal de Tesis, se le muestra una interfaz con todos los tribunales creados, escoge uno y procede a eliminarlo.	
Resultado Esperado: El tribunal seleccionado es eliminado de la lista de tribunales y el cambio se actualiza en la base de datos.	
Evaluación de la Prueba: Prueba satisfactoria.	

CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA

Nombre del caso de uso: Consultar Tribunal de Tesis

Caso de Prueba de Aceptación	
Código Caso de Prueba: HU-7_P11	Nombre Historia de Usuario: Gestionar tribunal de tesis.
Nombre de la persona que realiza la prueba: Leodan Cabrera Rodríguez	
Descripción de la Prueba: Prueba a la funcionalidad Consultar Tribunal de Tesis.	
Condiciones de Ejecución: El usuario tiene que estar registrado en el sitio y con los permisos correspondientes.	
Entrada / Pasos de ejecución: Una vez el usuario registrado en el sitio y con los permisos de acceder a las interfaces correspondientes, este puede escoger que acción realizar. Si el usuario escoge la opción de consultar tribunal, se muestra una interfaz donde aparecen todos los tribunales creados.	
Resultado Esperado: Se muestran todos los tribunales creados.	
Evaluación de la Prueba: Prueba satisfactoria.	

Nombre del caso de uso: Actualizar Tribunal de Tesis

Caso de Prueba de Aceptación	
Código Caso de Prueba: HU-7_P12	Nombre Historia de Usuario: Gestionar tribunal de tesis.
Nombre de la persona que realiza la prueba: Leodan Cabrera Rodríguez	
Descripción de la Prueba: Prueba a la funcionalidad Actualizar Tribunal de Tesis.	
Condiciones de Ejecución: El usuario tiene que estar registrado en el sitio y con los permisos correspondientes.	
Entrada / Pasos de ejecución: Una vez el usuario registrado en el sitio y con los permisos de acceder a las interfaces correspondientes, este puede escoger que acción realizar. Si el usuario escoge la opción de modificar tribunal, se muestra una interfaz donde aparecen todos los tribunales creados, escoge el que desea y procede a actualizarlo	
Resultado Esperado: Se actualiza el tribunal escogido y todos los cambios se registran en la base de datos.	

CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA

Evaluación de la Prueba: Prueba satisfactoria.

Nombre del caso de uso: Crear acta de evaluación.

Caso de Prueba de Aceptación	
Código Caso de Prueba: HU-9_P13	Nombre Historia de Usuario: Crear acta de evaluación.
Nombre de la persona que realiza la prueba: Leodan Cabrera Rodríguez	
Descripción de la Prueba: Prueba a la funcionalidad de Crear Acta de Evaluación.	
Condiciones de Ejecución: El usuario tiene que estar registrado en el sitio y con los permisos correspondientes.	
Entrada / Pasos de ejecución: Una vez el usuario registrado en el sitio y con los permisos de acceder a las interfaces correspondientes, este puede escoger que acción realizar. El usuario escoge la opción de crear una nueva acta. 1-El usuario da clic en el botón dejando campos obligatorios en blanco. 2-El usuario completa los campos correctamente.	
Resultado Esperado: 1-El sistema muestra un cartel: No se pueden dejar campos en blanco. 2-Con los datos correctos se crea la nueva acta y se almacena en la base de datos.	
Evaluación de la Prueba: Prueba satisfactoria.	

3.6 Conclusiones

En este capítulo se mostró la distribución física por nodos de la aplicación en el diagrama de despliegue. Además se crearon los diferentes diagramas de componentes representando las dependencias entre los componentes y se realizó la descripción de los casos de prueba de integración por caso de uso.

CONCLUSIONES GENERALES

CONCLUSIONES GENERALES

- La realización del presente trabajo de diploma demostró los problemas que presenta la gestión de las tesis de la Facultad 1 en la Universidad de la Ciencias Informáticas y se analizaron los distintos sitios existentes a nivel internacional y nacional para la gestión de tesis, estos estudios arrojaron que los sistemas existentes no cumplen con las funcionalidades que necesita la Facultad 1 por lo que se decide comenzar de cero.
- Las herramientas para realizar el proyecto con la calidad requerida y que estuvieran a tono con las disposiciones existentes, fueron elegidas sobre el uso de software libre y multiplataforma.
- En los centros estudiados el proceso se realiza de forma manual retardando así el proceso y generando más trabajo y gasto de recursos.
- La implementación del sistema resuelve las limitantes presentadas anteriormente y pone en manos de la Facultad 1 una herramienta que permite automatizar el proceso relacionado con la automatización de toda la información generada en el proceso de tesis de grado.
- El sistema cuenta con una interfaz Web amigable que muestra información referente a las tesis realizadas, además de permitir la administración del mismo.
- Se implementaron todas las funcionalidades propuestas obteniendo como resultado el cumplimiento de los objetivos trazados al inicio de la investigación.

RECOMENDACIONES

Luego de haber concluido el presente trabajo de diploma se recomienda:

- Estudiar la utilización e implementación de este sistema en otras facultades que aún no cuenten con uno o similar.
- Implementar funcionalidades que contemplen la solución para casos extremos como: el aplazamiento de una tesis, la suspensión de la misma a un estudiante, o la asignación de un profesor que no pertenezca a la universidad, en cualquiera de los roles, a una tesis.
- Mantener actualizada la información correspondiente al proceso de elaboración de tesis en la Facultad 1 de la Universidad de las Ciencias Informáticas.
- Estudiar nuevamente los procesos que se realizan en la facultad en busca de cambios o nuevas tareas que requieran automatización.

REFERENCIAS BIBLIOGRÁFICAS


1. **Sotolongo, Gilberto; Molina, Fernández.** Gestión de información, gestión del conocimiento y gestión de la calidad en las organizaciones. 2002. [Citado el: 6 de Noviembre de 2010]. Disponible en: [http://bvs.sld.cu/revistas/aci/vol10_5_02/aci04502.htm].
2. **Ministerio de Relaciones Exteriores de la República de Cuba (MINREX).** 2004. [Citado el: 6 de noviembre del 2010.]. Disponible en:[http://www.cubaminrex.cu/Sociedad_Informacion/Cuba_SI/Informatizacion.htm].
3. **Valarino,E.** Tesis a Tiempo. 2000. [Citado el: 8 de Noviembre de 2010]. Disponible en: [http://tesis.uci.cu/viewpage.php?page_id=11].
4. **Almazán, Blanca.** Automatización. 2008. [Citado el: 10 de Noviembre de 2010]. Disponible en: [<http://www.gestiopolis.com/administracion-estrategia/automatizacion-en-procesos-como-mejores-practicas.htm>].
5. **Diccionario de la Real Academia Española.** 2008. [Citado el: 10 de Noviembre de 2010]. Disponible en: [<http://www.rae.es/rae.html>].
6. **Carrión Maroto, Juan.** Introducción conceptual a la gestión del conocimiento. [Citado el: 11 de Noviembre del 2010]. Disponible en: [<http://www.gestiondelconocimiento.com/introduccion.htm>].
7. **Lamarca Lapuente, María Jesús.** Importancia de las bases de datos. 2009. [Citado el: 13 de Noviembre del 2010]. Disponible en: [<http://yerson94.blogspot.es/>]
8. **Colado Rodríguez. Aplicaciones Web.** 2003. [Citado el: 26 de diciembre del 2010].
9. **Cayetano Godoy, Manuel; Rubio, Manuel Angel.** Sistemas Gestores de Base de Datos [Citado el: 3 de Diciembre del 2010]. Disponible en: [<http://ciclo-asi.iespana.es/sbd/sbd.pdf>].
10. **Campos Paré, Rafael; Casillas Santillán, Luis Alberto; Costal Costa, Dolores.** Bases de Datos. 2005. [Citado el 3 de Diciembre del 2010]. [<http://www.uoc.edu/masters/oficiales/img/913.pdf>].
11. **González, Jhon Jairon.** [Citado el: 5 de Diciembre del 2010]. Disponible en: [http://www.jjairogm.com/index.php?option=com_content&view=article&id=16:servidor-web&catid=28:current-users&Itemid=44].

BIBLIOGRAFÍA

1. *Aja Quiroga, Lic. Lourdes. Gestión de información, gestión del conocimiento y gestión de la calidad en las organizaciones. Ciudad de La Habana: s.n., 2002.*
2. *Cordón Franco, Andrés. Sistemas de Gestión de Bases de Datos. Departamento de Ciencias de la Computación e Inteligencia Artificial (UNIVERSIDAD DE SEVILLA): s.n., 2005-2006.*
3. *Díaz Sardiñas, Adolfo. Modelo del Profesional y Objetivos de la carrera de Ingeniería en Ciencias Informáticas. Ciudad de la Habana: s.n., 2010.*
3. **Nevado Cabello, Victoria.** Introducción a Las Bases de Datos Relacionales. Madrid: s.n., 2010
4. Proyecto Estratégico 2008 - 2012 de la UCI. Objetivos de la UCI para el año 2010.
5. **Rivas, Lornel A; Pérez, María; Mendoza, Luis E; Grimán, Anna.** Herramientas de Desarrollo de Software: Hacia la Construcción de una Ontología. Venezuela.
6. **PRESSMAN, R. S.** *Ingeniería del Software Un enfoque práctico.* 2005. vol. 1,
7. POSTGRESQL. Sitio oficial de PostgreSQL, 1996-2008. [Disponible en: <http://www.postgresql.org/>]
8. **Habana, Por: Lázaro J. Blanco Encinosa. Doctor en Ciencias Económicas. Licenciado en Control Económico. Profesor Titular de Sistemas Informáticos. Facultad de Economía de la Universidad de La.** Apuntes para una historia de la Informática en Cuba. Consideraciones técnicas, organizativas y económicas.
9. **Solís Álvarez, Camilo Javier and Figueroa Díaz, Roberth Gustavo.** Universidad Técnica Particular de Loj. Metodologías Tradicionales vs. Metodologías Ágiles.
10. Tesis.uci. [En línea] 2006. <http://tesis.uci.cu/news.php>.
11. LibrosWeb.es. [En línea] <http://www.librosweb.es/ajax/capitulo1.html>
12. Características de PHP. [En línea] 04 06, 2009.

ANEXOS

Anexo 1 Crear Acta de evaluación.



Sistema de Gestión de Perfiles de Tesis

Inicio | [Crear contenido](#)

Titulo:

Fecha del Corte:

Lugar:

Nombre de la Comision:

Corte:

Perfiles:

Centro:

Dpto:

Proyecto:

Deficiencias Detectadas:

Sugerencias:

Evaluacion Concedida:

Dan constancia del acta

Presidente:

Secretario:

Vocal:

Estudiante(s):

Tutor(es):

leo1

- Mi cuenta
- Cerrar sesión

Enlaces del Tribunal

- Actas Creadas
- Crear Acta

Enlaces del Dpto

- Crear Tribunal
- Tribunales sin Tesis
- Ver Cronograma


Enlaces del Profesor

- Crear Perfil de Trabajo de Diploma
- Información del Profesor
- Perfiles Aprobados
- Perfiles a Rectificar
- Perfiles por Revisar

Copyright © 2009 Sistema de Gestión de Perfiles de Tesis. All Rights Reserved.

Anexo #1: Crear Acta de evaluación.

Anexo 2 Crear Perfil de Trabajo de Diploma



Sistema de Gestión de Perfiles de Tesis

Inicio | [Crear contenido](#)

Título del Perfil: *

Identificación

Curso Académico: *

Título: Año 52 del Triunfo de la Revolución ▾

Tutor(es): *

⊕ ○

⊕ ○

Cantidad Autores: *

Usuario del Estudiante:

⊕ ○

⊕ ○

Llenar solo después que el perfil es aprobado....

Clasificación: *

--Seleccione-- ▾

Clasificación según el área de Desarrollo: *

--Seleccione-- ▾

Síntesis del Autor: *

Contexto del Problema: *

Problema a Resolver: *

Elementos del Diseño Teórico

Objetivo General: *

Tareas a Cumplimentar: *

Resultados Esperados: *

Centro al que Pertenece: *

GEITEL ▾

Bibliografía Fundamental: *

Perfil Disponible: *


Sí

No

Copyright © 2009 Sistema de Gestión de Perfiles de Tesis. All Rights Reserved.

Anexo 2: Crear Perfil de Trabajo de Diploma.

Anexo 3 Crear Tribunal .



Sistema de Gestión de Perfiles de Tesis

Inicio | [Crear contenido](#)

Título: *

Integrantes

Presidente: *

Usuario del Presidente: *

Secretario: *

Usuario del Secretario: *

Vocal: *

Usuario del Vocal: *

Tesis Disponibles:

Opciones Disponibles:

Opciones Seleccionadas:

Copyright © 2009 Sistema de Gestión de Perfiles de Tesis. All Rights Reserved.

Anexo 3: Crear Tribunal.

Anexo 4 Tribunal 1.



Sistema de Gestión de Perfiles de Tesis

leo1

- [Mi cuenta](#)
- [Cerrar sesión](#)

Inicio

[Ver](#) [What links here](#) [Editar](#)

Tribunal 1

Vie, 05/27/2011 - 12:13 | leo1

Enlaces del Tribunal

- [Actas Creadas](#)
- [Crear Acta](#)

Enlaces del Dpto

- [Crear Tribunal](#)
- [Tribunales sin Tesis](#)
- [Ver Cronograma](#)

Enlaces del Profesor

- [Crear Perfil de Trabajo de Diploma](#)
- [Información del Profesor](#)
- [Perfiles Aprobados](#)
- [Perfiles a Rectificar](#)
- [Perfiles por Revisar](#)

Integrantes

Presidente:
Juan Alberto

Usuario del Presidente:
leodan

Secretario:
Yoel Cabrera

Usuario del Secretario:
leodan

Vocal:
Sergio Azcuy

Usuario del Vocal:
leodan

Copyright © 2009 Sistema de Gestión de Perfiles de Tesis. All Rights Reserved.

Anexo 4: Tribunal 1