



INSTITUTO SUPERIOR MINERO METALÚRGICO

“Dr. Antonio Núñez Jiménez”.

Facultad de Metalurgia - Electromecánica

Moa, Holguín

SISTEMA DE PREGRADO

MÓDULO GESTIÓN Y SEGUIMIENTO DE LOS TRABAJOS DE DIPLOMA

EN EL ISMM

Trabajo de Diploma

Presentado en Opción al Título de Ingeniera en Informática

Autora: Viviana Nicot Hernández.

Tutor: Ing. Dabiel González Ramos.

Moa, Cuba

Julio, 2010

DECLARACIÓN DE AUTORÍA

Declaro que soy la única autora de este trabajo y autorizo al Instituto Superior Minero Metalúrgico “Dr. Antonio Núñez Jiménez” para que hagan el uso que estimen pertinente con este trabajo.

Para que así conste firmo la presente a los _____ días del mes _____ del 2010.

Viviana Nicot Hernández

Nombre completo del primer autor

Ing. Dabiel González Ramos

Nombre completo del primer tutor

OPINIÓN DEL USUARIO DEL INFORME TÉCNICO

El Trabajo de Diploma, titulado “**Sistema de Gestión y Seguimiento de los Trabajos de Diploma en el ISMM**”, fue realizado en nuestra entidad: Instituto Superior Minero Metalúrgico de Moa. Se considera que, en correspondencia con los objetivos trazados, el trabajo realizado le satisface:

_ Totalmente

_ Parcialmente en un _____ %

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

Y para que así conste, se firma la presente a los _____ días del mes de _____ del año 2010.

Nombre del representante de la entidad

Cargo

Firma

Cuño

OPINIÓN DEL TUTOR DEL TRABAJO DE DIPLOMA

Título: Sistema de Gestión y Seguimiento de los Trabajos de Diploma en el ISMM.

Autor: Viviana Nicot Hernández.

El tutor del presente Trabajo de Diploma considera que durante su ejecución el estudiante mostró las cualidades que a continuación se detallan.

Ing. Dabiel González Ramos

Nombre completo del primer tutor

Fecha: _____

PENSAMIENTO

Mejor es adquirir sabiduría que oro preciado; y adquirir inteligencia vale más que la plata.

Proverbios 16:16

DEDICATORIA

A mi mamá: a quien más quiero, por haberme apoyado en mis decisiones, por darme su confianza, cariño y amor. Este es mi regalo por tu esfuerzo en mi educación.

A Freddy: la persona que más me ama y me apoyó en los momentos más difíciles.

A mi bebé: a quien voy a amar con todo el corazón.

AGRADECIMIENTOS

A Dios por guiarme en mis pasos y no abandonarme nunca, por permitir que este sueño se realizara y por rodearme de tantas personas maravillosas que me dieron aliento y cariño.

A mi mamá y a Erlis por su amor y comprensión. Por su esfuerzo y empeño en mi educación. Gracias por su ayuda y su cariño.

A Freddy, una persona muy especial en mi vida que supo darme todo su amor, su cariño, su comprensión, su apoyo incondicional, sus consejos y ayuda. Alguien que me apoyó mucho en mi vida y sobre todas las cosas me amó. A ti muchísimas gracias porque gran parte de esta gloria te la debo a ti.

A mis suegros Freddy y Odalis, por haberme dado todo su amor y apoyo incondicional.

A mi familia, por todo su afecto y aliento en todo momento. A mis tíos Roiler, Rebeca, Raquel (quin), Pancho, Darcy y Roberto (beto) y a mis primos Robertico, Adriano, Adriana, Yuselis, Darbet, Yurina, Teresita, por todo su apoyo y comprensión. A todos gracias por su cariño y amor.

A mis amigas y al grupo Info 2005, quienes estuvieron ahí para animarme e impulsarme a seguir adelante: a Yanet, Tamara, Katy y Maidelis.

Al claustro de profesores que me acompañó durante estos cinco años de estudio.

A todas aquellas personas que han estado a mi lado no sólo en este período de universidad sino durante toda mi vida, a todos esos amigos que me han brindado su mano, gracias sencillamente por el granito de arena que me proporcionaron para realizar esta ilusión. A todos ustedes sinceramente:

Gracias.

RESUMEN

Con el perfeccionamiento constante de las Tecnologías de la Información y las Comunicaciones (TIC) han surgido nuevos escenarios docentes. La Educación Superior siempre ha estado acorde con el uso de tecnología en pro de mejorar la calidad del proceso de Enseñanza y Aprendizaje; además el desarrollo acelerado en las últimas décadas de la informática ha permitido, con su aplicación en las Universidades, alcanzar un eslabón superior para este tipo de enseñanza. El Trabajo de Diploma por muchos años ha constituido una forma de adquirir mayor dominio de los métodos investigativos y técnicos característicos de la profesión.

Con el desarrollo de esta investigación se propone la implementación de un sistema que permita gestionar y controlar de manera ágil y segura toda la información referente al proceso de Trabajos de Diploma en el ISMM.

Para llevar a cabo este trabajo se realizó un análisis de la bibliografía utilizada para el desarrollo de aplicaciones Web y de las herramientas a utilizar en la construcción de las mismas; y también se investigó de manera profunda acerca de la metodología de Ingeniería de Software que se siguió para la construcción del software que se propone como solución de la problemática encontrada.

ABSTRACT

With constant improvement of information and communication technologies (ICT), new teaching scenarios have emerged. Higher education has always been in accordance with the use of technologies in order to improve the quality of the teaching process. On the other hand, the fast development of information technologies in the last decades has allowed reaching a higher step in this type of learning, with its application in universities. During years, the research work has been a way of acquiring more knowledge regarding research methods and specific profession features.

This research work is aimed at implementing a system for managing and controlling, in expedite and safe manner, all the information regarding Research Work Process at the ISMM.

An analysis of the bibliography used for the development of web applications and tools, used in the elaboration of these applications, was conducted to carry out this research work. The methodology of software engineering, used to elaborate the software that is proposed as a solution for the problems found, was also studied very carefully.

ÍNDICE

INTRODUCCIÓN	1
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA	7
1.1 INTRODUCCIÓN.....	7
1.2 DESCRIPCIÓN DEL OBJETO DE ESTUDIO.....	7
1.2.1 ¿Qué es la investigación científica?.....	7
1.2.2 ¿Qué es un Trabajo de Diploma?.....	8
1.2.3 Importancia de los Trabajos de Diploma.....	8
1.3 LOS TRABAJOS DE DIPLOMA EN EL INSTITUTO SUPERIOR MINERO METALÚRGICO.....	9
1.3.1 Flujo actual de los procesos	9
1.3.2 Análisis crítico de la ejecución de los procesos	10
1.3.3 Propuesta de la solución	10
1.4 SISTEMAS AUTOMATIZADOS EXISTENTES VINCULADOS AL CAMPO DE ACCIÓN.....	11
1.5 TENDENCIAS Y TECNOLOGÍAS ACTUALES.....	11
1.5.1 Metodologías ágiles aplicadas al desarrollo de software	11
1.5.1.1 XP (Extreme Programming – Kent Beck) (Programación Extrema)	13
1.5.1.2 SCRUM.....	15
1.5.1.3 Crystal (Alistair Cockburn).....	15
1.5.1.4 DSDM (DSDM Consortium) (Método dinámico del desarrollo de los sistemas)	16
1.5.1.5 Feature Driven Development (Jeff de Luca, Peter Coad, Together Soft)	17
1.5.1.6 Adaptive Software Development (Jim Highsmith) (Desarrollo adaptante del software).....	18
1.5.2 Lenguajes de programación para aplicaciones Web.....	19
1.5.2.1 PHP.....	19
1.5.2.2 Perl.....	20
1.5.2.3 ASP.....	20
1.5.2.4 JSP.....	21
1.5.3 Gestores de Bases de Datos	21
1.5.3.1 SQL Server.....	22
1.5.3.2 PostgreSQL.....	22
1.5.3.3 Mysql.....	23
1.5.4 Frameworks de desarrollo	24
1.5.4.1 JSF(Java Server Faces)	24
1.5.4.2 .NET Framework	24
1.5.4.3 CodeIgniter.....	25
1.6 TECNOLOGÍAS Y HERRAMIENTAS USADAS PARA EL DESARROLLO DE LA APLICACIÓN.....	26
1.6.1 Metodología a utilizar.....	26
1.6.1.1 Fases de la metodología SXP	27
1.6.2 Lenguaje de programación	28
1.6.3 Gestor de Base de Datos	29
1.6.4 Framework de desarrollo	29
1.6.5 Otras herramientas necesarias.....	30
1.6.5.1 Macromedia Dreamweaver 8.....	30

1.6.5.2	<i>Herramientas CASE</i>	30
1.7	CONCLUSIONES DEL CAPÍTULO.....	31
CAPÍTULO 2: ANÁLISIS Y DISEÑO		33
2.1	INTRODUCCIÓN.....	33
2.2	PLANIFICACIÓN – DEFINICIÓN.....	33
2.2.1	<i>Concepción del sistema</i>	33
2.2.2	<i>Modelo de Historias de usuario del negocio</i>	36
2.2.3	<i>Lista de reserva del producto (LRP)</i>	37
2.2.4	<i>Historias de usuario</i>	40
2.2.5	<i>Valoración de riesgos</i>	41
2.2.6	<i>Diseño</i>	43
2.2.6.1	<i>Modelo de diseño</i>	43
2.3	CONCLUSIONES DEL CAPÍTULO.....	44
CAPÍTULO 3: DESARROLLO DEL SISTEMA		45
3.1	INTRODUCCIÓN.....	45
3.2	DESARROLLO.....	45
3.2.1	<i>Plan de realeses</i>	45
3.2.2	<i>Cronograma de Producción</i>	46
3.2.3	<i>Tarjetas CRC</i>	48
3.2.4	<i>Tareas por Historias de Usuario</i>	50
3.3	CONCLUSIONES DEL CAPÍTULO.....	51
CAPÍTULO 4: PRUEBAS		52
4.1	INTRODUCCIÓN.....	52
4.2	PRUEBAS.....	52
4.2.1	<i>Pruebas de aceptación</i>	52
4.3	CONCLUSIONES DEL CAPÍTULO.....	53
CAPÍTULO 5: ESTUDIO DE FACTIBILIDAD		54
5.1	INTRODUCCIÓN.....	54
5.2	EFFECTOS ECONÓMICOS.....	54
5.2.1	<i>Efectos directos</i>	55
5.2.2	<i>Efectos indirectos</i>	55
5.2.3	<i>Efectos externos</i>	55
5.2.4	<i>Intangibles</i>	55
5.3	BENEFICIOS Y COSTOS INTANGIBLES EN EL PROYECTO.....	56
5.4	FICHA DE COSTO.....	57
5.5	CONCLUSIONES DEL CAPÍTULO.....	60
CONCLUSIONES GENERALES		61
RECOMENDACIONES		62
BIBLIOGRAFÍA		63
GLOSARIO DE TÉRMINOS		66
ANEXO 1		I

ÍNDICE	DE	TABLAS
TABLA 1	ROLES QUE INTERVIENEN EN EL DESARROLLO DEL SISTEMA	35
TABLA 2	DEFINICIÓN DE ACTORES DEL SISTEMA	37
TABLA 3	LISTA DE RESERVA DEL PRODUCTO	38
TABLA 4	DESCRIPCIÓN DE LA HISTORIA DE USUARIO <GESTIONAR TEMA DE TESIS>	40
TABLA 5	LISTA DE RIESGOS	41
TABLA 6	PLAN DE REALESES	46
TABLA 7	CRONOGRAMA DE PRODUCCIÓN	47
TABLA 8	TARJETA CRC <CTEMASTESIS>	49
TABLA 9	TARJETA CRC <CCORTETESIS>	49
TABLA 10	TARJETA CRC <CTUTOREAR>	49
TABLA 11	DESCRIPCIÓN DE LA TAREA DE INGENIERÍA < APROBAR TEMA DE TESIS >	50
TABLA 12	CASO DE PRUEBA DE ACEPTACIÓN DE LA HISTORIA DE USUARIO <GESTIONAR TEMAS DE TESIS>	53
TABLA 13	EXPLICACIÓN DEL GRÁFICO DE PUNTO DE EQUILIBRIO DE SOLUCIONES	60

ÍNDICE DE FIGURA

FIGURA 1 DIAGRAMA DE PAQUETES	43
FIGURA 2 GRÁFICO DE PUNTO DE EQUILIBRIO DE SOLUCIONES.....	59
FIGURA 3 MODELO DE DATOS	I



INTRODUCCIÓN

En el mundo contemporáneo, la utilización de las Tecnologías de la Información y las Comunicaciones (TIC), han impulsado de forma acelerada el desarrollo científico-técnico de los países, en la industria, la economía, la salud y la educación, aún cuando estos avances tecnológicos sigan siendo un privilegio de los países del primer mundo o desarrollados, mientras que en los países menos desarrollados se hace imprescindible disponer de una estrategia nacional para su utilización como herramientas de desarrollo económico, social y cultural.

Su avance y su influencia en todas las esferas de la sociedad ha permitido no solo el incremento en los resultados de la ciencia, la producción y los servicios sino que también se refleja en la forma de actuar y pensar de los individuos, donde lo logrado hasta hoy era catalogado como inalcanzable, utópico y, a veces, imposible imaginarlo.

Las tecnologías de la información potencian la generalización del aprendizaje a lo largo de toda la vida al hacer de la educación un proceso más fácil, individualizado y asequible; y contribuyen de manera ostensible a la generalización del conocimiento.

En la educación, como muchas otras actividades, su uso creciente ha venido dictado por la evolución de éstas. En Cuba se han aplicado a la educación desde hace bastante tiempo, pero fue la aparición de las computadoras personales a comienzos de los años 80 el hito que permitió que la informática fuera un recurso barato y con grandes prestaciones, accesible a todos. Las mejoras continuas del hardware y otras tecnologías han extendido y acelerado su uso.



Lo cierto es que han permitido la transformación de los programas de formación de recursos humanos, han diversificado la educación y han surgido nuevos escenarios docentes, y la utilización de nuevos medios de enseñanza desde los videos conferencias, tele conferencias, aulas virtuales, sistemas de softwares educativos, multimedias y cursos a distancia. Así mismo, esta incorporación de las comunicaciones hace prever nuevas posibilidades y desarrollos en un futuro próximo.

La Educación Superior siempre ha estado acorde con el uso de tecnología en pro de mejorar la calidad del proceso de Enseñanza y Aprendizaje. El desarrollo acelerado en las últimas décadas de la informática ha permitido, con su aplicación en las Universidades, alcanzar un eslabón superior para este tipo de enseñanza.

Es interés del Ministerio de Educación de Cuba, llevar a todos sus Centros Universitarios, los nuevos adelantos de las técnicas más actualizadas de la información y las comunicaciones, al servicio de profesores y estudiantes, con el objetivo de lograr un egresado con mente abierta y gran creatividad, capaz de enfrentar los retos que ofrece la sociedad de estos tiempos.

Como centro de educación superior el Instituto Superior Minero Metalúrgico de Moa situado en la provincia de Holguín , trabaja por incorporar las facilidades y ventajas del uso de las nuevas tecnologías en la vida cotidiana, en la formación de profesionales, en las investigaciones científicas, entre otras actividades que contribuyan a elevar el nivel técnico y humano que promueven a crear una cultura en su utilización y con el empeño de buscar opciones exitosas para dar soluciones a las dificultades materiales.

Por muchos años el Trabajo de Diploma les ha permitido a los estudiantes adquirir un mayor dominio de los métodos investigativos y técnicos



característicos de la profesión y este se realiza fundamentalmente en una de las esferas de actuación del futuro profesional.

El contenido de los Trabajos de Diploma se adecua a los objetivos establecidos en los planes de estudio. Además, constituye una evaluación de culminación de la carrera.

Generalmente los estudiantes buscan propuestas de Trabajos de Diplomas al arribar al 5to año. Muchas veces esta acción se dificulta ya que hay que ver personalmente al profesor responsable para poder elegir un tema de tesis y luego el estudiante tiene que ver a su tutor para programar encuentros con este. Actualmente en el ISMM todo este proceso de seguimiento de los Trabajos de Diploma se realiza manualmente (proposición de temas, publicación de fechas de cortes evaluativos, etc.) ocasionando que este trabajo sea engorroso para los involucrados y que se obstaculice la comunicación entre el estudiante y su tutor.

El trabajo de investigación a realizar pretende dar solución a la **situación problémica** expuesta anteriormente, por lo que se plantea como **problema**: La ausencia de un sistema informático que permita la gestión y el seguimiento de los Trabajos de Diploma en el ISMM.

Por lo antes expuesto se tiene como **objeto de estudio**: Gestión y Seguimiento de los Trabajos de Diploma en el ISMM.

Y como **campo de acción**: Sistema informático para la gestión y el seguimiento de los Trabajos de Diploma en el ISMM.

Para poder llevar a cabo esta investigación se plantea como **hipótesis**: el desarrollo de un sistema informático para el control de los Trabajos de Diploma en el ISMM favorecerá el seguimiento sobre cada uno de éstos y propiciará un



ahorro de tiempo tanto para los estudiantes como para los profesores involucrados en el proceso.

Como **objetivo general** para dar solución al problema se propone: Diseñar e Implementar una aplicación Web que sea capaz de gestionar y controlar la información correspondiente a los Trabajos de Diploma en el ISMM.

Debido a esto se plantean los siguientes **objetivos específicos**:

- Estudiar cómo se lleva a cabo el proceso de seguimiento de Trabajos de Diploma en el ISMM.
- Realizar una búsqueda bibliográfica sobre sistemas informáticos vinculados al proceso de Trabajos de Diploma.
- Diseñar y documentar el Sistema de Gestión y Seguimiento de los Trabajos de Diploma en el ISMM.
- Implementar el Sistema de Gestión y Seguimiento de los Trabajos de Diploma en el ISMM.

Para que se cumplan estos objetivos es necesario plantear las siguientes **tareas**:

- Entrevistar a diferentes entidades (J'Dptos, Profesores, Secretarías) involucradas para fomentar como se desarrolla el proceso de seguimiento de los Trabajos de Diploma en cada una.
- Buscar información nacional e internacional sobre los Trabajos de Diploma.
- Realizar el análisis del sistema gestión y seguimiento de los Trabajos de Diploma en el ISMM.
- Realizar el diseño e implementar el sistema.
- Realizar pruebas al sistema.
- Desarrollar el manual de usuario.



Para cumplimentar estas tareas se han empleado métodos empíricos y teóricos de la investigación científica.

Entre los métodos **empíricos** que se utilizaron para poder recopilar información se pueden citar:

La **Observación Científica**: se utiliza para poder ver como funciona el proceso de gestión y seguimiento de los Trabajos de Diploma en el ISMM.

La **Entrevista**: permite recopilar información necesaria para valorar la situación actual del problema lo cual permitió realizar un análisis y determinar los principales requisitos del sistema.

Entre los métodos **teóricos** que se utilizaron se pueden citar:

Histórico-Lógico: Para conocer el fenómeno que se estudia en sus antecedentes y tendencias actuales, lo cual permite establecer las bases teóricas que sustentan la investigación. También para reflejar de forma lógica la esencia, necesidad y el comportamiento del proceso de Trabajos de Diploma en el instituto y el desarrollo del software.

Análisis y Síntesis: permitió resumir los aspectos más importantes de la bibliografía consultada vinculada con el problema tratado y otras investigaciones relacionadas con el objeto de estudio; y se realizó la descomposición de cada uno de los requisitos del sistema para un mejor análisis.

Hipotético-Deductivo: permite realizar el debido análisis para el posterior desarrollo del sistema que dará solución al problema existente.



Este trabajo tiene como **aporte práctico** el Diseño e Implementación del Sistema Informático para la Gestión eficiente de los Trabajos de Diploma en el ISMM.

El presente trabajo consta de introducción, 5 capítulos, conclusiones generales, recomendaciones, bibliografía, glosario de términos y anexos.

Capítulo 1 Fundamentación Teórica: se ofrece una breve descripción de diferentes conceptos imprescindibles que le dan base a la investigación. Se brinda además una panorámica de los sistemas automatizados existentes vinculados al campo de acción y así como algunos elementos importantes de las herramientas a utilizar para la implementación del sistema.

Capítulo 2 Análisis y Diseño: se aborda los aspectos funcionales para el desarrollo del sistema y se definen por medio de las historias de usuario cuales son los procesos fundamentales de la aplicación.

Capítulo 3 Desarrollo del sistema: contiene los aspectos relacionados con la implementación del sistema. Se describen las tareas de ingeniería asociadas a cada historia de usuario, se presenta la técnica de tarjetas CRC y se desarrolla el cronograma de producción en el cual se refleja cada una de las actividades a desarrollar por el equipo de trabajo.

Capítulo 4 Pruebas: este capítulo está dedicado a las pruebas que le realiza el cliente, por medio de las pruebas de aceptación, al software. Estas se les realizan a cada una de las historias de usuario para comprobar y validar las funcionalidades del sistema.

Capítulo 5 Estudio de factibilidad: en este capítulo se realizará el estudio de factibilidad del sistema mediante la Metodología Costo-Efectividad (Beneficio).



CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

1.1 Introducción

En este capítulo se aborda de forma general los conceptos asociados al dominio del problema. También se hace una breve descripción de las herramientas y tecnologías utilizadas para el desarrollo de la aplicación así como de la metodología a utilizar y sus principales características.

1.2 Descripción del objeto de estudio

1.2.1 ¿Qué es la investigación científica?

La investigación científica es una de las tres funciones de la Universidad y por la importancia que reviste en la formación de los estudiantes es necesario trabajar en el mejoramiento continuo del componente investigativo. **(MARTÍN W., F.; E. BALBIS COLL,)**

El trabajo investigativo de los estudiantes se integra, como un sistema, con los tipos de actividades académica y laboral; y contribuye, fundamentalmente, al desarrollo de la iniciativa, la independencia cognoscitiva, el fomento de intereses por esta actividad y la apropiación del método científico-investigativo; todo lo cual posibilita el desarrollo de la creatividad de los estudiantes y a la formación de convicciones materialista-dialécticas.



Además, propicia el desarrollo de las habilidades para el uso eficiente y actualizado de las fuentes de información, de los idiomas extranjeros, de los métodos y técnicas de la computación, de los análisis económicos, de la conservación del medio ambiente, control de calidad, normalización y metrología.

1.2.2 ¿Qué es un Trabajo de Diploma?

Un Trabajo de Diploma o tesis es una afirmación cuya veracidad ha sido argumentada, demostrada o justificada de alguna manera. Generalmente enuncia una proposición científica, un axioma o un hecho demostrable. **[Tesis, Wikipedia, la enciclopedia libre, 2010]**

Su objetivo básico es el de informar a través de un soporte material, a un jurado calificador, acerca de un nuevo conocimiento que se ha logrado establecer a partir de una investigación.

En su objetivo se encuentra su razón de ser, es necesario cumplir con lo establecido por una disciplina para recibir un título, bien sea éste profesional, de licenciatura o doctoral ya que permite poner en práctica los conocimientos adquiridos en clase y obtener mejores resultados.

Además el Trabajo de Diploma es el tipo de trabajo investigativo que les permite a los estudiantes adquirir un mayor dominio de los métodos investigativos y técnicos característicos de la profesión y este se realiza fundamentalmente en una de las esferas de actuación del futuro profesional.

1.2.3 Importancia de los Trabajos de Diploma

La tesis o Trabajo de Diploma constituye un trabajo académico, resultado de un proceso de formación, que se inicia desde el ingreso del estudiante a la universidad y se liga necesariamente a la investigación.



La persona que realiza una tesis, tiene la capacidad de reflexionar sobre los problemas no sólo inmediatos que se encuentran a su alrededor, sino también sobre la relación que tienen éstos con los problemas globales.

La importancia a realizar una tesis radica en que ofrece al universitario la oportunidad de efectuar una lectura de la realidad que rompa con los parámetros de una sola visión, implicando el desafío de reconocer la necesidad de desarrollar una respuesta ante una situación problemática que lleva consigo el riesgo de equivocarse.

El proceso de formación que se lleva a cabo en la universidad debe posibilitar que el estudiante desarrolle formas de pensar, la realidad desde otros ángulos de manera tal que no quede atrapado en una sola lógica de pensamiento. Ello se expresa en actos de conciencia crítico-constructivos de nuevas realidades.

1.3 Los Trabajos de Diploma en el Instituto Superior Minero Metalúrgico

1.3.1 Flujo actual de los procesos

Un estudiante universitario al arribar al 5to año de la carrera recibe o busca propuestas de desarrollo de algún Trabajo de Diploma. Estas son hechas por profesores del centro, generalmente de su propia carrera, y encaminadas al perfil del profesional que ha recibido el estudiante durante sus estudios.

Cada estudiante elige el tema que desee pero lo tiene que hacer saber de forma personal. Luego que ya tiene su tema de tesis tiene que ver al tutor correspondiente con el tema para recibir orientaciones sobre lo que debe hacer, programar encuentros con éste y para que realice revisiones a su trabajo de tesis.



También en este proceso se realizan varias evaluaciones para saber como va el desarrollo de la investigación, más bien es un Proceso de Defensa de Trabajos de Diplomas de los cuales se da a conocer: fecha, el tipo de evaluación que se va a realizar (1er Corte, 2do Corte, etc.) y los objetivos a evaluar.

1.3.2 Análisis crítico de la ejecución de los procesos

El proceso descrito en el epígrafe anterior se realiza manualmente ocasionando que este trabajo sea arduo para todos los involucrados en el proceso. Esto trae consigo muchas consecuencias, como por ejemplo, si el estudiante no ve con tiempo al profesor responsable de los temas de tesis puede ocasionar demora en la selección de éstos, otra dificultad es que a veces se imposibilita la comunicación entre el estudiante y su tutor ya que éste puede que se encuentre muy ocupado o que no lo pueda ver. Durante este proceso otro problema que existe es que el profesor debe ir guardando toda la información a revisar al estudiante, lo que provoca muchas veces desajustes en los cambios realizados por parte del autor y además otro inconveniente es que las informaciones correspondientes a este proceso de tesis se dan a conocer por e-mail pero este no siempre es una vía segura.

1.3.3 Propuesta de la solución

Como se ha podido apreciar en los epígrafes anteriores todo este proceso llevado a cabo de forma manual provoca que el progreso de este sea más lento y también que haya pérdida de información.

Es por ello, que para dar solución a la problemática existente, se propone implementar un módulo, dentro del Sistema de Pregrado, que permita registrar toda la información referente a la gestión de los Trabajos de Diploma en el ISMM y que posibilite hacer un seguimiento de los mismos.



1.4 Sistemas automatizados existentes vinculados al campo de acción

Actualmente no se ha dado a conocer la existencia de algún sistema informático vinculado al campo de acción en el ámbito nacional. Se conoce que la Universidad de Ciencias Informáticas (UCI) trabaja en la implementación de un ERP universitario que incorpora un módulo de gestión de tesis.

Nuestro instituto sólo cuenta con una base de datos hecha en el gestor de base de datos CDS/ISIS for Windows (versión 1.3) en la que se realiza un registro de todos los Trabajos de Diplomas desarrollados en el centro cada curso escolar los cuales están organizados por especialidad.

1.5 Tendencias y Tecnologías actuales

1.5.1 Metodologías ágiles aplicadas al desarrollo de software

Las Metodologías Ágiles o “ligeras” constituyen un nuevo enfoque en el desarrollo de software, mejor aceptado por los desarrolladores de e-projects (proyectos) que las metodologías convencionales (ISO-9000, CMM, etc.) debido a la simplicidad de sus reglas y prácticas, su orientación a equipos de desarrollo de pequeño tamaño, su flexibilidad ante los cambios y su ideología de colaboración.

Las metodologías ágiles se entienden como un conjunto de metodologías para desarrollo de software surgidas en la década de los 90.

Lo ágil se define (por los mismos agilistas) como la habilidad de responder de forma versátil al cambio para maximizar los beneficios.



Características de las metodologías ágiles

De manera general, las metodologías ágiles pueden explicarse a través de los siguientes 4 principios fundamentales que son a su vez los propósitos del Manifiesto Ágil.

1. Al individuo y las interacciones del equipo de desarrollo sobre el proceso y las herramientas. Dado que el proceso de desarrollo es creativo, no es posible pensar que las personas funcionen respondiendo a órdenes o procesos rígidos.
2. Desarrollar software que funciona más que conseguir una buena documentación. Puesto que si el software no funciona la documentación no vale de nada. A nivel interno puede haber documentación, pero sólo la necesaria y a nivel externo lo que el cliente requiera.
3. La colaboración con el cliente más que la negociación de un contrato. Supone que la satisfacción del cliente con el producto será mayor, mientras exista una conversación y realimentación continua entre éste y la empresa.
4. Responder a los cambios más que seguir estrictamente un plan. Puesto que si un proyecto de software no es capaz de adaptarse a los cambios fracasara, especialmente en productos de gran envergadura. La estrategia de planificación se basa en: planes detallados para las próximas semanas, planes aproximados para los próximos meses y muy generales para plazos mayores.

A nivel mundial existen varias metodologías para el desarrollo de software que encajan bajo el estandarte de ágil. Mientras todas ellas comparten muchas características, también hay algunas diferencias significativas. Dentro de las más conocidas podemos citar:

- XP (Extreme Programming – Kent Beck)



- SCRUM (Jeff Sutherland, Ken Schwaber)
- Crystal (Alistair Cockburn)
- DSDM (DSDM Consortium)
- Feature Driven Development (Jeff deLuca, Peter Coad, Together Soft)
- Adaptive Software Development (Jim Highsmith)

1.5.1.1 XP (Extreme Programming – Kent Beck) (Programación Extrema)

XP es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en desarrollo de software, promoviendo el trabajo en equipo, preocupándose por el aprendizaje de los desarrolladores, y propiciando un buen clima de trabajo. XP se basa en realimentación continua entre el cliente y el equipo de desarrollo, comunicación fluida entre todos los participantes, simplicidad en las soluciones implementadas y coraje para enfrentar los cambios.

Valores que promueve la metodología X.P

Comunicación

Al desarrollar sistemas software con una u otra metodología se determina que la comunicación es una pieza importante para un buen desarrollo de los proyectos, ya que se realiza un intercambio de ideas entre jefes, entre el personal que manejará el sistema, terceras personas que aportan, el equipo de desarrollo y el equipo de trabajo.

La comunicación con el cliente es fluida ya que forma parte del equipo de desarrollo. Una mala comunicación con el cliente puede representar en ocasiones (requisitos mal comprendidos) procesos mal definidos o inexactos. Además este decide qué características tienen prioridad y siempre debe estar disponible para solucionar dudas.



Simplicidad

La simplicidad es la base de la programación extrema. Cuando se desarrolla con interfaces y lógica simple, los sistemas mejoran la portabilidad, manejo, mantenimiento, capacitación y rapidez, siendo un punto a favor para la adquisición o desarrollo de dicho sistema. Aplicando la simplicidad junto con la autoría del código se asegura que cuanto más grande se haga el proyecto, se conocerá más y mejor el sistema completo.

Retroalimentación (feedback)

Este es un punto importante para resaltar, aumenta mucho la socialización y comunicación entre los distintos equipos relacionados en el proyecto, es recomendado manejar una retroalimentación constante y oportuna en todas las fases del proyecto de forma rápida y clara.

Coraje o Valentía

Ser valientes en el desarrollo de proyectos software es asumir riesgos, enfrentarse a ellos y continuar al límite junto a esta actitud hasta terminar un sistema. Es un valor importante y arriesgado que gracias a la pertenencia que se maneja en XP permite la toma de decisiones importantes para el buen desarrollo de los proyectos software.

Esta metodología se basa en:

- **Pruebas Unitarias:** se basa en las pruebas realizadas a los principales procesos, de tal manera que adelantando en algo hacia el futuro, se pueden hacer pruebas de las fallas que pudieran ocurrir. Es como si se adelantara a obtener los posibles errores.
- **Refabricación:** se basa en la reutilización de código, para lo cual se crean patrones o modelos estándares, siendo más flexible al cambio.



- **Programación en pareja:** una particularidad de esta metodología es que propone la programación en pares, la cual consiste en que dos desarrolladores participen en un proyecto en una misma estación de trabajo. Cada miembro lleva a cabo la acción que el otro no está haciendo en ese momento.

1.5.1.2 SCRUM

SCRUM define un marco para la gestión de proyectos, que se ha utilizado con éxito durante los últimos 10 años. Sus principales características se pueden resumir en dos. El desarrollo de software se realiza mediante iteraciones, denominadas sprints, con una duración de 30 días. El resultado de cada sprint es un incremento ejecutable que se muestra al cliente. La segunda característica importante son las reuniones a lo largo del proyecto, entre ellas se destaca la reunión diaria de 15 minutos del equipo de desarrollo para coordinación e integración.

Características

- Equipos auto dirigidos.
- Utiliza reglas para crear un entorno ágil de administración de proyectos.
- No prescribe prácticas específicas de ingeniería.
- Los requerimientos se capturan como ítems de la lista reserva del producto.
- El producto se construye en una serie de sprints de un mes de duración.

1.5.1.3 Crystal (Alistair Cockburn)

Crystal Methodologies se trata de un conjunto de metodologías para el desarrollo de software caracterizadas por estar centradas en las personas que componen el equipo y la reducción al máximo del número de artefactos producidos. El desarrollo de software se considera un juego cooperativo de invención y comunicación, limitado por los recursos a utilizar. El equipo de desarrollo es un factor clave, por lo que se deben invertir esfuerzos en mejorar sus habilidades y destrezas, así como tener políticas de trabajo en equipo



definidas. Estas políticas dependerán del tamaño del equipo, estableciéndose una clasificación por colores, por ejemplo Crystal Clear (Amarillo) (3 a 8 miembros) y Crystal Orange (Naranja) (25 a 50 miembros).

Características:

- La cantidad de detalle necesario en la documentación de los requerimientos, diseño y planeamiento varía según las circunstancias del proyecto.
- Quizás no sea posible eliminar todos los work products (productos del trabajo) y las notas promisorias inmediatas tales como los documentos de requerimientos, diseño y los planes para el proyecto, pero pueden reducirse a una comunicación corta, enriquecedora, e informal entre el equipo.
- El equipo ajusta continuamente sus convenciones de trabajo para adaptarse a:
 - Las personalidades particulares del equipo.
 - El ambiente local de trabajo actual.
 - Las particularidades de las asignaciones específicas.

1.5.1.4 DSDM (DSDM Consortium) (Método dinámico del desarrollo de los sistemas)

Define el marco para desarrollar un proceso de producción de software. Nace en 1994 con el objetivo de crear una metodología RAD unificada. Sus principales características son: es un proceso iterativo e incremental y el equipo de desarrollo y el usuario trabajan juntos. Propone cinco fases: estudio viabilidad, estudio del negocio, modelado funcional, diseño y construcción, y finalmente implementación. Las tres últimas son iterativas, además de existir realimentación a todas las fases.



Características:

- **Must have:** Debe tener. Son los requerimientos fundamentales del sistema. De éstos, el subconjunto mínimo ha de ser satisfecho por completo.
- **Should have:** Debería tener. Son requerimientos importantes para los que habrá una resolución en el corto plazo.
- **Could have:** Podría tener. Podrían quedar fuera del sistema si no hay más remedio.
- **Want to have but won't have this time around:** Se desea que tenga, pero no lo tendrá esta vuelta. Son requerimientos valorados, pero pueden esperar.

1.5.1.5 Feature Driven Development (Jeff de Luca, Peter Coad, Together Soft)

Define un proceso iterativo que consta de 5 pasos. Las iteraciones son cortas (hasta 2 semanas). Se centra en las fases de diseño e implementación del sistema partiendo de una lista de características que debe reunir el software.

FDD consiste en un conjunto de “mejores prácticas” que distan de ser nuevas pero se combinan de manera original. Las prácticas canónicas son:

- Modelado de objetos del dominio, resultante en un framework (marco) cuando se agregan los rasgos.
- Desarrollo por rasgo. Hacer simplemente que las clases y objetos funcionen no refleja lo que el cliente pide. El seguimiento del progreso se realiza mediante examen de pequeñas funcionalidades descompuestas y funciones valoradas por el cliente.
- Propiedad individual de clases (código).
- Equipos de rasgos, pequeños y dinámicamente formados builds regulares. Siempre se tiene un sistema disponible. Los builds forman la base a partir de la cual se van agregando nuevos rasgos.



- Administración de configuración. Permite realizar seguimiento histórico de las últimas versiones completas de código fuente.
- Reporte de progreso. Se comunica a todos los niveles organizacionales necesarios.

1.5.1.6 Adaptive Software Development (Jim Highsmith) (Desarrollo adaptante del software)

Sus principales características son: iterativo, orientado a los componentes software más que a las tareas y tolerante a los cambios. El ciclo de vida que propone tiene tres fases esenciales: especulación, colaboración y aprendizaje. En la primera de ellas se inicia el proyecto y se planifican las características del software; en la segunda se desarrollan las características y finalmente en la tercera se revisa su calidad, y se entrega al cliente. La revisión de los componentes sirve para aprender de los errores y volver a iniciar el ciclo de desarrollo.

ASD presupone que las necesidades del cliente son siempre cambiantes. La iniciación de un proyecto involucra definir una misión para él, determinar las características y las fechas y descomponer el proyecto en una serie de pasos individuales, cada uno de los cuales puede abarcar entre cuatro y ocho semanas. Los pasos iniciales deben verificar el alcance del proyecto; los tardíos tienen que ver con el diseño de una arquitectura, la construcción del código, la ejecución de las pruebas finales y el despliegue.

Aspectos claves de ASD son:

1. Un conjunto no estándar de “artefactos de misión” (documentos para tí y para mí), incluyendo una visión del proyecto, una hoja de datos, un perfil de misión del producto y un esquema de su especificación
2. Un ciclo de vida, inherentemente iterativo.
3. Cajas de tiempo, con ciclos cortos de entrega orientados por riesgo.



1.5.2 Lenguajes de programación para aplicaciones Web

Los lenguajes de programación Web posibilitan la interacción y personalización de la información con el usuario. Existen varios lenguajes, entre ellos se encuentran Java, PHP, PERL, ASP, JSP, entre otros.

A continuación se muestran los lenguajes de programación y tecnologías más importantes de la Web en la actualidad y la fundamentación teórica de las mismas.

1.5.2.1 PHP

PHP que debe sus siglas, originalmente, a “Personal Home Page” y en la actualidad a “Hypertext Preprocessor” es un lenguaje interpretado de propósito general muy usado, diseñado especialmente para el desarrollo de sistemas Web el cual puede ser embebido dentro de código html. Es un lenguaje gratuito, rápido, multiplataforma y compatible con los gestores de bases de datos más comúnmente empleados, que ha devenido en uno de los lenguajes de programación Web más usados en Internet.

La comunidad de desarrolladores que colaboran en las mejoras a este lenguaje ofrecen una amplia librería de funciones que cubre desde cálculos matemáticos complejos hasta tratamiento de conexiones de red, y cobertura documental a las nuevas implementaciones. Es innegable como PHP ha incrementado sus adeptos desde su surgimiento hasta la actualidad.

El uso de PHP ha proliferado muy rápidamente en los últimos años. El número creciente de desarrolladores han hallado en este lenguaje la mejor variante para las implementaciones Web y éste constituye una amenaza para otros lenguajes y tecnologías propietarias que se sentían muy seguras en el mercado de las aplicaciones Internet.

Ventajas de PHP:

- Es un lenguaje multiplataforma.



- Leer y manipular datos desde diversas fuentes, incluyendo datos que pueden ingresar los usuarios desde formularios HTML.
- 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.
- Permite las técnicas de Programación Orientada a Objetos.
- Permite crear los formularios para la Web.
- Biblioteca nativa de funciones sumamente amplia.

1.5.2.2 Perl

Es el acrónimo de Practical Extracting and Reporting Language. Es un lenguaje de programación muy utilizado para construir aplicaciones CGI (Computer Graphic Interface) para el Web. Perl se trata de un lenguaje de programación muy práctico para extraer información de archivos de texto y generar informes a partir del contenido de los ficheros.

Es un lenguaje gratuito. Antes estaba muy asociado a la plataforma Unix, pero en la actualidad está disponible en otros sistemas operativos como Windows. Este lenguaje está inspirado a partir de lenguajes como C, sh, y sed (algunos provenientes de los sistemas Unix), pero está enfocado a ser más práctico y fácil que estos últimos. Una diferencia fundamental de Perl con respecto a los otros lenguajes es que no limita el tamaño de los datos con los que trabaja, el límite lo pone la memoria que en ese momento se encuentre disponible.

1.5.2.3 ASP

Es el acrónimo de Active Server Pages Es la tecnología desarrollada por Microsoft para la creación de páginas dinámicas del servidor. ASP se escribe



en la misma página Web, utilizando el lenguaje Visual Basic Script o JScript (JavaScript de Microsoft).

Actualmente se ha presentado la segunda versión de ASP, el ASP.NET, que comprende algunas mejoras en cuanto a posibilidades del lenguaje y rapidez con la que funciona. ASP.NET tiene algunas diferencias en cuanto a sintaxis con el ASP, de modo que se ha de tratar de distintas maneras uno de otro. Para implementarlo es necesario montar en el Servidor la Plataforma .NET.

ASP.NET está basado en el Entorno Común de Ejecución de Lenguajes (Common Language Runtime, o "CLR"), el poder y la flexibilidad de la plataforma completa está disponible para los desarrolladores. Además permite el uso de una gran variedad de lenguajes de programación.

1.5.2.4 JSP

Es el acrónimo de Java Server Pages. Es una tecnología orientada a crear páginas Web con programación en Java. Con JSP se pueden crear aplicaciones Web que se ejecuten en variados servidores Web, de múltiples plataformas, ya que Java es en esencia un lenguaje multiplataforma. Las páginas JSP están compuestas de código HTML/XML mezclado con etiquetas especiales para programar scripts de servidor en sintaxis Java.

El motor de las páginas JSP está basado en los servlets de Java, programas en Java destinados a ejecutarse en el servidor , aunque el número de desarrolladores que pueden afrontar la programación de JSP es mucho mayor, dado que resulta mucho más sencillo de aprender que los servlets.

1.5.3 Gestores de Bases de Datos

Entiéndase por Sistema Gestor de Base de Datos (SGBD) un sistema que permite el procesamiento de la información almacenada en bases de datos, así como el acceso a la misma por varios usuarios en forma concurrente mediante la realización de peticiones.



Tiene entre sus funciones principales garantizar la integridad, seguridad y el control centralizado de los datos que se almacenan, facilitando la manipulación eficaz de la información, independizando los datos de los programas de aplicación y minimizando las redundancias.

1.5.3.1 SQL Server

Microsoft SQL Server, propietario de Microsoft, pertenece a la familia de los sistemas de administración de base de datos, operando en una arquitectura cliente/servidor de gran rendimiento. Su desarrollo fue orientado para hacer posible manejar grandes volúmenes de información, y un elevado número de transacciones. SQL Server es una aplicación completa que realiza toda la gestión relacionada con los datos. El servidor sólo tiene que enviarle una cadena de caracteres (la sentencia SQL) y esperar a que le devuelvan los datos.

SQL Server permite la creación de procedimientos almacenados, los cuales consisten en instrucciones SQL que se almacenan dentro de una base de datos de SQL Server, realizados en lenguaje SQL, se trata de procedimientos que se guardan semicompilados en el servidor y que pueden ser invocados desde el cliente. Se ejecutan más rápido que instrucciones SQL independientes.

SQL Server puede manejar perfectamente bases de datos de Terabytes con millones de registros y funciona sin problemas con miles de conexiones simultáneas a los datos, sólo depende de la potencia del hardware del equipo en el que esté instalado y solamente corre sobre Windows NT- 2000 Server.

1.5.3.2 PostgreSQL

PostgreSQL es un Sistema de Gestión de Bases de Datos Objeto-Relacionales (ORDBMS) que ha sido desarrollado de varias formas desde 1977. Comenzó como un proyecto denominado Ingres en la Universidad Berkeley de California.



Ingres fue más tarde desarrollado comercialmente por la Relational Technologies/Ingres Corporation.

En 1986 otro equipo dirigido por Michael Stonebraker de Berkeley continuó el desarrollo del código de Ingres para crear un sistema de bases de datos objeto-relacionales llamado Postgres. En 1996, debido a un nuevo esfuerzo de código abierto y a la incrementada funcionalidad del software, Postgres fue renombrado a PostgreSQL, tras un breve periplo como Postgres95. El proyecto PostgreSQL sigue actualmente un activo proceso de desarrollo a nivel mundial gracias a un equipo de desarrolladores y contribuidores de código abierto.

PostgreSQL está considerado como el gestor de base de datos de software libre más avanzado del mundo. Es un sistema objeto – relacional pues incluye aspectos del paradigma orientada a objetos, tales como la herencia, tipos de datos, funciones, restricciones, reglas e integridad transaccional, aunque no llega a ser un gestor con orientación a objetos pura.

Proporciona un gran número de características que normalmente sólo se encontraban en las bases de datos comerciales como Oracle.

1.5.3.3 Mysql

MySQL es un sistema de administración de Base de Datos. Opera en una arquitectura cliente/servidor. Es un proyecto “open source”. Permite la fácil conectividad, alta velocidad de respuesta a solicitudes y gran seguridad, por ello se utiliza para acceder a Bases de Datos desde Internet.

MySQL es muy rápido, confiable y fácil de usar, es multiplataforma, multiusuario y permite elaborar consultas con el robusto SQL, además no tiene valor monetario, es un software que se puede adquirir libremente, la licencia es completamente libre.



El lenguaje PHP es altamente compatible con MySQL, por el amplio conjunto de comandos definidos para el tratamiento de este.

1.5.4 Frameworks de desarrollo

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

1.5.4.1 JSF(Java Server Faces)

Es un framework para aplicaciones basadas en Web que simplifica el desarrollo de interfaces de usuario en aplicaciones Java EE. JSF usa JavaServer Page (JSP) como la tecnología que permite hacer el despliegue de las páginas. JSF incluye:

1. Un conjunto de APIs para representar componentes de una interfaz de usuario y administrar su estado, manejar eventos, validar entrada, definir un esquema de navegación de las páginas y dar soporte para internacionalización y accesibilidad.
2. Un conjunto por defecto de componentes para la interfaz de usuario.
3. Dos librerías de etiquetas personalizadas para JavaServer Pages que permiten expresar una interfaz JavaServer Faces dentro de una página JSP.
4. Un modelo de eventos en el lado del servidor.
5. Administración de estados.

1.5.4.2 .NET Framework

El .NET Framework (traducido como “Marco de Trabajo”) es el componente fundamental de la plataforma Microsoft .NET, necesario tanto para poder desarrollar aplicaciones como para poder ejecutarlas luego en entornos de prueba o producción.



.NET Framework es el modelo de programación de la plataforma .NET para generar, implementar y ejecutar aplicaciones Web, aplicaciones de cliente inteligente y Servicios Web XML (Lenguaje de marcado extensible).

Los componentes clave de .NET Framework son Common Language Runtime y la biblioteca de clases .NET Framework, que incluye ADO.NET, ASP.NET y los formularios Windows Forms. .NET Framework proporciona un entorno de ejecución administrado, un desarrollo e implementación simplificada y la integración con una gran variedad de lenguajes de programación.

Es una nueva plataforma informática que simplifica el desarrollo de aplicaciones en un entorno altamente distribuido como es Internet.

1.5.4.3 CodeIgniter

CodeIgniter es un poderoso framework para PHP que facilita la escritura de código repetitivo, y a comparación de otros frameworks como CakePHP, Symphony o Zend Framework.

Su principal objetivo es ayudar a que los desarrolladores, puedan realizar proyectos mucho más rápido que creando toda la estructura desde cero, a través de proveer un rico conjunto de librerías para tareas comúnmente necesarias, tanto como una simple interface y estructura lógica para acceder a estas librerías. CodeIgniter le permite concentrarse creativamente en su proyecto minimizando el volumen de código necesario para una tarea determinada.

Hay que destacar que CodeIgniter es más rápido que muchos otros entornos. Incluso en una discusión sobre entornos de desarrollo con PHP, Rasmus Lerdorf (el creador de PHP) expresó que le gustaba CodeIgniter "porque es rápido, ligero y parece poco un entorno".



Desde un punto de vista técnico y arquitectónico, Codelgniter fue creado con los siguientes objetivos:

- **Instanciación Dinámica.** En Codelgniter, los componentes son cargados y las rutinas ejecutadas sólo cuando se pide, en vez de globalmente. No se hacen suposiciones por el sistema acerca de que puede necesitar además de los recursos mínimos del núcleo, así que el sistema es muy liviano por defecto. Los eventos que se desencadenan por la petición HTTP y los controladores y vistas que diseñe determinarán que es invocado.
- **Poco Acoplamiento.** Acoplamiento es el grado que los componentes de un sistema dependen entre ellos. Mientras menos componentes dependan de otro, más reusable y flexible el sistema se vuelve. Nuestra meta era un sistema con muy poco acoplamiento.
- **Singularidad del Componente.** Singularidad es el grado que más componentes tienen un propósito en el que enfocarse más estrecho. En Codelgniter, cada clase y sus funciones son altamente autónomas para permitir máxima utilidad.

1.6 Tecnologías y Herramientas usadas para el desarrollo de la aplicación

1.6.1 Metodología a utilizar

Seleccionar la metodología adecuada permite en gran medida guiar el proceso de desarrollo, encaminado a obtener un software con calidad, flexible y escalable, que cumpla con las necesidades de los usuarios.

Para llevar a cabo el presente trabajo se tiene como propuesta utilizar la unión de 2 metodologías explicadas anteriormente SCRUM Y XP, tomando lo mejor de cada una de ellas. Esta unión da como resultado la metodología SXP.



SCRUM y XP se llevan muy bien y son complementarios, por lo que es buena idea utilizar ambos simultáneamente. La metodología SCRUM se usará para la planificación de los proyectos que usarán métodos ágiles como metodología para su proceso de desarrollo, pues SCRUM es una forma de gestionar proyectos de software, no es una metodología de análisis, ni de diseño, es una metodología de gestión del trabajo. La metodología XP se usará para la parte de desarrollo ya que es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en el desarrollo de software, promoviendo el trabajo en equipo, preocupándose por el aprendizaje de los desarrolladores, y propiciando un buen clima de trabajo.

¿Por qué SXP?

1. Flexibilidad ante los cambios para maximizar los beneficios.
2. Se realizan entregas periódicas y frecuentes que funcionen, en un período corto de tiempo.
3. Buen diseño y calidad técnica.
4. La simplicidad es algo básico.
5. El cliente forma parte del equipo de desarrollo.

1.6.1.1 Fases de la metodología SXP

- 1 **Fase de Definición:** se centra sobre el qué, aquí se intenta que información ha de ser procesada, que función y rendimiento se desea, que comportamiento del sistema, que interfaces va a ser establecidas, que duración se necesita para definir un sistema correcto, en fin, han de identificarse los requisitos claves del sistema y del software. Cuenta con tres tareas principales: ingeniería de sistemas o de información, planificación del proyecto y análisis de los requisitos.
- 2 **Fase de Desarrollo:** se centra en el cómo, se intenta definir cómo han de diseñarse las estructuras de datos, como ha de implementarse la función dentro de una arquitectura de software, como han de



implementarse los detalles de procedimentales, como han de caracterizarse interfaces, cómo ha de traducirse el diseño en un lenguaje de programación y cómo ha de realizarse la prueba. Cuenta con tres tareas: diseño del software, generación de código y prueba del software.

- 3 **Entrega:** En esta fase se realiza la entrega del software y su documentación, generándose aquellos documentos que son imprescindibles para el entrenamiento y entendimiento del producto.

- 4 **Fase de Mantenimiento:** Se centra en el cambio que va asociado a la corrección de errores, a las adaptaciones requeridas a medida que va evolucionando el entorno del software y a cambios debido a las mejoras producidas por los requisitos cambiantes del cliente. Tipos de cambios: Corrección, Adaptación, Mejora, Prevención.

1.6.2 Lenguaje de programación

PHP

Luego de hacer el análisis entre varios lenguajes de programación, se decide utilizar el PHP embebido en el código HTML ya que:

1. Está soportado en la mayoría de las plataformas de Sistemas Operativos, lo que nos permite la creación de una aplicación multiplataforma.
2. Permite crear los formularios para la Web.
3. Capacidad de conexión con la mayoría de los gestores de base de datos que se utilizan en la actualidad.
4. El PHP no tiene costo oculto, o sea que cuando se adquiere incluye un sinnúmero de bibliotecas que proporcionan el soporte para la mayoría de las aplicaciones Web, por ejemplo e-mail, generación de ficheros PDF y otros. En caso de que no se tengan las bibliotecas, éstas se pueden encontrar gratis en Internet.



1.6.3 Gestor de Base de Datos

PostgreSQL

Como gestor de Base de Datos, se utilizará PostgreSQL, que además de las características mencionadas anteriormente, también incluye:

- Control de Concurrencia Multi-Versión es la tecnología que PostgreSQL usa para evitar bloqueos innecesarios, está considerado mejor que el bloqueo a nivel de fila porque un lector nunca es bloqueado por un escritor.
- PostgreSQL soporta integridad referencial, la cual es utilizada para garantizar la validez de los datos que gestiona.
- Es altamente extensible: soporta operadores, funciones, métodos de acceso y tipos de datos definidos por el usuario.
- Mantiene una ruta a todas las transacciones realizadas por los usuarios de la base de datos y es capaz entonces de manejar.
- Permite instalación ilimitada. Frecuentemente las base de datos comerciales son instaladas en más servidores de lo que permite la licencia
- Tiene la posibilidad de manejar grandes bases de datos, se han llegado a mantener mediante PostgreSQL bases de datos de hasta de 60GB.

1.6.4 Framework de desarrollo

CodeIgniter

Como framework de desarrollo se escoge CodeIgniter ya que entre sus características podemos encontrar las siguientes:

- Sistema Basado en Modelo-Vista-Controlador, que permite una buena separación entre lógica y presentación. Esto es particularmente bueno para proyectos en los cuales los diseñadores están trabajando con sus archivos de plantilla, ya que el código en esos archivos será mínimo.



- Extremadamente Liviano. El núcleo del sistema sólo requiere unas pocas pequeñas librerías. Las librerías adicionales son cargadas dinámicamente a pedido, basado en sus necesidades para un proceso dado, así que el sistema base es muy delgado y bastante rápido.
- Clases de base de datos llenas de características con soporte para varias plataformas.
- Formulario y Validación de Datos.
- Realmente rápido.
- Es extensible: El sistema puede ser fácilmente extendido a través del uso de plugins y librerías asistentes, o a través de extensión de clases o ganchos del sistema.

1.6.5 Otras herramientas necesarias

1.6.5.1 Macromedia Dreamweaver 8

Como se pretende implementar una aplicación Web para confeccionar la propuesta de este trabajo, se hace necesario tener en cuenta la utilización de un editor de páginas Web.

Para esta función la elección no ha sido muy difícil, ya que la herramienta de creación de sitios Web más utilizada en la actualidad es el Macromedia Dreamweaver 8. Con esa herramienta se podrá desarrollar cualquier sitio Web personal con características de sitio profesional y utilizar casi todos los recursos de la Web, así como realizar aplicaciones que se ejecuten en servidor y vinculaciones dinámicas de datos; además de contar con un soporte para aplicaciones PHP y utilización de bases PostgreSQL. También cuenta con un amplio soporte para la creación y utilización de CSS (Cascading Style Sheets) para lograr un diseño fácil y óptimo.

1.6.5.2 Herramientas CASE

A medida que los sistemas que hoy se construyen se tornan más y más complejos, las herramientas de modelado con UML ofrecen muchos beneficios



para todos los involucrados en un proyecto, por ejemplo, administrador del proyecto, analistas, arquitectos, desarrolladores y otros. Las herramientas CASE de modelado con UML nos permiten aplicar la metodología de análisis y diseño orientados a objetos y abstraernos del código fuente, en un nivel donde la arquitectura y el diseño se tornan más obvios y más fáciles de entender y modificar. Cuanto más grande es un proyecto, es más importante utilizar una herramienta CASE.

Algunos ejemplos de herramientas CASE:

- System Architect, herramientas CASE para Análisis y Diseño, incluye técnicas estructuradas y orientadas a objetos.
- Win A&D, herramientas CASE para Análisis y Diseño, incluye técnicas estructuradas y orientadas a objetos.
- Power Designer 7.0: herramienta CASE de Análisis y Diseño incluye capacidades de generación relacional y con orientación a objetos.
- Rational Rose, herramienta CASE para Análisis y Diseño basándose en el Proceso Unificado de Rational (RUP).
- Visual Paradigm, herramienta CASE para Análisis y Diseño, utiliza el Lenguaje Unificado de Modelado (UML).

Selección de la herramienta CASE a utilizar

La herramienta CASE seleccionada fue el Case Studio2 ya que es una herramienta profesional para realizar el diseño de la base de datos, facilitando herramientas para la creación de diagramas de relación, modelado de datos y gestión de estructuras (Es compatible con los siguientes sistemas gestores de base de datos: MySQL y Postgres).

1.7 Conclusiones del Capítulo

Al concluir este capítulo se ha podido apreciar por qué surge la necesidad de diseñar un sistema capaz de dar respuesta al problema de la gestión y seguimiento de los Trabajos de Diploma en el ISMM. Además explica los



principales aspectos relacionados con el tema brindando una visión más amplia del sistema que se pretende desarrollar.



CAPÍTULO 2: ANÁLISIS Y DISEÑO

2.1 Introducción

En este capítulo se abordará la fase de planeación y diseño, la cual es muy importante ya que se generan las principales plantillas que se encuentran relacionadas con la concepción inicial del sistema. En esta se especificarán los requisitos funcionales y no funcionales del sistema y quedarán plasmadas las historias de usuarios (user histories o HU).

2.2 Planificación - Definición

La fase de Planificación – Definición es la primera que define la metodología SXP. En esta fase se generan todos los documentos que se encuentran relacionados con la concepción inicial del sistema, así como la definición del mismo. También se incluyen algunos que están vinculados a la primera parte de los procesos de Ingeniería de Software, tales como los relacionados con el negocio, los requisitos y el diseño. Y como también hay una parte de planificación en esta fase, se incluyen aquellos documentos que están relacionados con la estimación inicial de esfuerzos, y la valoración de los riesgos.

2.2.1 Concepción del sistema

La plantilla de Concepción del sistema es el primer documento que se genera en la fase de Planificación-Definición. En este además de reflejarse la visión general del producto a implementar se da a conocer los diferentes roles que intervendrán en el desarrollo del software así como las responsabilidades a



asumir en dicho proceso. Se recoge también cuales serán las herramientas utilizadas en la implementación de la aplicación y los motivos de la necesidad del desarrollo del software.

Para comprender cuales son los datos que se recogerán en esta plantilla, se muestra la misma a continuación.

Clasificación del proyecto

Aplicación Web.

Tipo de proyecto

Nacional.

Resumen

El presente documento tratará temas relacionados con la Gestión y el Seguimiento de los Trabajos de Diploma en el ISMM, las razones por las cuales surge el proyecto y qué se pretende solucionar con el mismo, así como los roles que intervendrán en el desarrollo del trabajo por parte de los clientes y desarrolladores.

Palabras claves: Trabajo de Diploma, Tesis de Pregrado.

Surgimiento

El desarrollo acelerado en las últimas décadas de la informática ha permitido, con su aplicación en las Universidades, alcanzar un eslabón superior para este tipo de enseñanza. El Instituto Superior Minero Metalúrgico trabaja por incorporar las facilidades y ventajas del uso de las nuevas tecnologías en la vida cotidiana , es por ello que se ha estudiado el proceso de Gestión y Seguimiento de los Trabajos de Diploma, ya que dicho proceso se realiza de forma manual ocasionando que este trabajo sea engorroso para los involucrados en este proceso y además que se obstaculice la comunicación



entre el estudiante y su tutor, por lo que surgió la idea de que era necesario desarrollar una herramienta informática que ayudara en la Gestión y el Seguimiento de los Trabajos de Diploma.

¿Qué es?

El Módulo de Gestión y Seguimiento de los Trabajos de Diploma no es más que una vía de solución para gestionar y controlar de manera ágil y segura toda la información referente al proceso de Trabajos de Diploma en el ISMM, basado en un modelo de software libre. La idea es crear un sistema que les permita a los estudiantes de 5to año conocer los temas de tesis disponibles según su especialidad. Además éste le da la posibilidad, tanto a un estudiante como a un profesor, de publicar algún tema de tesis luego de haber sido aprobado por el departamento de la carrera. También ofrece la ventaja a los profesores de poder ser tutor de alguna tesis propuesta en el sitio y en el caso de que ya tenga diplomante podrá realizar revisiones desde el sistema al Trabajo de Diploma. Esta aplicación posibilita que se visualice la información correspondiente a los cortes de tesis lo que permite que el estudiante se mantenga informado.

Metodología a utilizar

SCRUM-XP (SXP)

Roles

Tabla 1 Roles que intervienen en el desarrollo del sistema

Rol	Nombre
Gerente	Ing. Dabiel González Ramos
Analista	Viviana Nicot Hernández
Diseñador	Viviana Nicot Hernández
Desarrollador	Viviana Nicot Hernández
Cliente	Instituto Superior Minero Metalúrgico



	(ISMM)
--	--------

Misión

La aplicación Web que se desea desarrollar tiene como misión constituir la herramienta básica y fundamental que permita gestionar y hacer un seguimiento de los Trabajos de Diploma en el ISMM.

Visión

Se espera contar con un portal Web interactivo que tenga una interfaz sencilla, la cual permita que se pueda interactuar de manera fácil con el sistema.

Herramientas utilizadas

- Metodología SXP.
- PHP.
- PostgreSQL.
- CodeIgniter.
- Dreamweaver.
- Case Studio2.

2.2.2 Modelo de Historias de usuario del negocio

Mediante el Modelo de Historias de usuario del negocio se concretan las características específicas del negocio, así como la forma en que interactúa el sistema con los clientes y viceversa. Como se está trabajando con metodologías ágiles, el modelo de negocio es diferente al ya conocido en el proceso unificado, ya que en este caso se trabaja con historias de usuarios, en vez de con casos de uso. Pero independientemente de los cambios técnicos que puedan existir, el negocio se modela igual en cualquier metodología.

A continuación se muestran cuales son los actores que intervienen en el negocio.



Tabla 2 Definición de actores del sistema

Actor	Descripción
Jefe de Carrera	Tendrá derecho a visualizar la información necesaria en el sistema.
Jefe Departamento	Es el encargado de aprobar los temas de tesis antes de su publicación en el sistema y tendrá acceso al sistema para poder visualizar la información que este desee.
Profesores	Es usuario del sistema y podrá tener acceso a la información publicada en el mismo.
Estudiantes	Es usuario del sistema y podrá interactuar con el mismo.

2.2.3 Lista de reserva del producto (LRP)

La plantilla de Lista de reserva del producto es el primer artefacto generado en la etapa de Captura de requisitos y está conformada por una lista priorizada que define el trabajo que se va a realizar en el proyecto.

Cuando un proyecto comienza es difícil tener bien claros todos los requerimientos del producto. Se crea una lista de requerimientos la cual puede crecer y modificarse a medida que se obtienen más conocimientos acerca del producto y del cliente, con la restricción de que sólo puede cambiarse entre iteraciones. El objetivo es asegurar que el producto definido al terminar la lista es el más correcto, útil y competitivo posible y para esto la lista debe acompañar los cambios en el entorno y el producto.

Para comprender cuales son los datos que se recogerán en esta plantilla, se muestra la misma gráficamente.



Tabla 3 Lista de reserva del producto

Asignado a	Ítem *	Descripción	Estimación	Estimado por
		Prioridad	Muy Alta	
Viviana Nicot Hernández	1	Un estudiante puede proponer un tema de tesis.	1	Analista
Viviana Nicot Hernández	2	Un profesor puede proponer un tema de tesis.	1	Analista
Viviana Nicot Hernández	3	Los temas de tesis antes de ser propuestos deben ser aprobados por el departamento.	1	Analista
		Prioridad	Alta	
Viviana Nicot Hernández	4	Un estudiante acepta un tema de tesis propuesto por un profesor.	3	Analista
Viviana Nicot Hernández	5	Un profesor accede a tutorear algún tema de tesis propuesto por un estudiante.	1	Analista
		Prioridad	Media	
Viviana Nicot Hernández	6	Un estudiante brinda la información a revisar por el tutor.	1	Analista
Viviana Nicot Hernández	7	El profesor realiza revisiones a la tesis que tutorea.	2	Analista
Viviana Nicot Hernández	8	Publicar en el sistema toda la información correspondiente a los cortes de tesis (objetivos a evaluar, fecha).	2	Analista
		Prioridad	Baja	
Requisitos no Funcionales				
Viviana Nicot Hernández	9	Apariencia o Interfaz Externa Diseño sencillo permitiendo que	14 semanas	



		se pueda interactuar de manera fácil con el sistema. Su interfaz agradable logra captar la confianza de los usuarios así como su identificación con los colores y formatos en que realizan sus acciones.		
Viviana Nicot Hernández	10	Usabilidad El sistema en general se desarrolla con el objetivo de facilitar el trabajo manual realizado hasta el momento por lo que se espera que cuente con un alto nivel de usabilidad por los usuarios debido a que constituye una forma más fácil de gestionar los Trabajos de Diploma. El sistema garantizará un acceso fácil y rápido a los usuarios.	14 semanas	
Viviana Nicot Hernández	11	Rendimiento Como la aplicación está concebida para un ambiente cliente/servidor los tiempos de respuestas deben ser generalmente rápidos al igual que la velocidad de procesamiento de la información. Estará implementado sobre una tecnología Web, facilitando su uso a través de la red.	14 semanas	
Viviana Nicot	12	Soporte	14 semanas	



Hernández		La base de datos que utiliza el sistema como medio de almacenamiento de la información esta soportada sobre un gestor de bases de datos PostgreSQL.		
Viviana Nicot Hernández	13	Portabilidad El sistema trabaja sobre sistema Windows y Linux.	14 semanas	

2.2.4 Historias de usuario

Las historias de usuario son la técnica utilizada en XP para especificar los requisitos del software, lo que equivaldría a los casos de uso en el proceso unificado. Las mismas son escritas por los clientes como las tareas que el sistema debe hacer y su construcción depende principalmente de la habilidad que tenga el cliente para definirlas.

Las historias de usuario guían la construcción de las pruebas de aceptación, elemento clave en XP y son utilizadas para estimar tiempos de desarrollo. En este sentido, sólo proveen detalles suficientes para hacer una estimación razonable del tiempo que llevará implementarlas. En el momento de implementar una historia de usuario, se debe detallar a través de la comunicación con el cliente. Estas son la base para las pruebas funcionales.

Para comprender cuales son los datos que se recogerán en esta plantilla, se muestra la misma gráficamente.

Tabla 4 Descripción de la historia de usuario <Gestionar tema de tesis>

Historia de Usuario	
Número: 1	Nombre Historia de Usuario: Gestionar tema de tesis.



Modificación de Historia de Usuario Número: ninguna.	
Usuario: Viviana Nicot Hernández.	Iteración Asignada: 1
Prioridad en Negocio: Muy Alta	Puntos Estimados: 3
Riesgo en Desarrollo: Bajo	Puntos Reales: 2,5
Descripción: Permitir que un estudiante o un profesor publiquen un tema de tesis.	
Observaciones: Se hace referencia a las funcionalidades 1,2 y 3 enumeradas en la Lista de reserva del producto.	
Prototipo de interface	

Para ver las demás Plantillas de Historias de Usuarios, ir al Expediente de Proyecto: 04 - Plantilla de Historias de Usuarios.

2.2.5 Valoración de riesgos

La plantilla de lista de riesgos, es el documento que se genera de la actividad de valoración de riesgos. En ella se definen los posibles riesgos que actuarán sobre el proceso de desarrollo de software, así como la estrategia trazada para mitigarlos, además de un plan de contingencia que describe que curso seguirán las acciones si el riesgo se materializa.

Para comprender cuales son los datos que se recogerán en esta plantilla, se muestra la misma gráficamente.

Tabla 5 Lista de riesgos

Riesgo	Tipos de riesgos	Impacto	Descripción	Probabilidad	Efectos	Mitigación del riesgo
--------	------------------	---------	-------------	--------------	---------	-----------------------



1. Escaso dominio de las herramientas y tecnologías a utilizar.	Tecnológico.	Puede atrasar el plazo de entrega del software	Estudiando las herramientas y tecnologías a utilizar en el desarrollo del software se puede llegar a la conclusión que estas son las más factibles a emplear por sus características; muchas de ellas son nuevas, es decir, que se tiene poco dominio de su uso por lo que se considera un riesgo para la investigación.	Media	Serias	1. Intenso estudio de las bibliografías que tratan estas tecnologías. 2. Entrevista con personas capacitadas en el tema.
2. Falta de comunicación con los clientes.	Organización	Pobre captura de requisitos.	Los clientes tienen muchas responsabilidades, por lo que no disponen de mucho tiempo para realizar reuniones y así determinar los posibles requisitos del sistema.	Media	Serias	1. Planificación de reuniones previas con los clientes

Plan de Contingencia.

- **Riesgo # 1**

En caso de que no se logre mitigar este riesgo se debe tomar como una solución alternativa utilizar herramientas y tecnologías que se conozcan.



- **Riesgo # 2**

Si no se logra el contacto con los clientes se deberá considerar estudiar el negocio por otras vías.

2.2.6 Diseño

2.2.6.1 Modelo de diseño

La plantilla del Modelo de diseño, es el documento que se genera del Diseño con las metáforas, donde se debe diseñar la solución más simple que pueda funcionar y ser implementada en un momento determinado del proyecto. En XP no se enfatiza la definición temprana de una arquitectura estable para el sistema. Dicha arquitectura se asume de forma evolutiva y los posibles inconvenientes que se generarían por no contar con ella explícitamente en el comienzo del proyecto se solventan con la existencia de una metáfora.

El sistema es definido mediante una metáfora o un conjunto de metáforas compartidas por el cliente y el equipo de desarrollo. Una metáfora es una historia compartida que describe cómo debería funcionar el sistema.

Teniendo en cuenta las características anteriores, se define un esbozo inicial del diseño del sistema, sin entrar en especificaciones, ni detalles, sólo lo que se necesita para hacer un primer entregable del sistema.

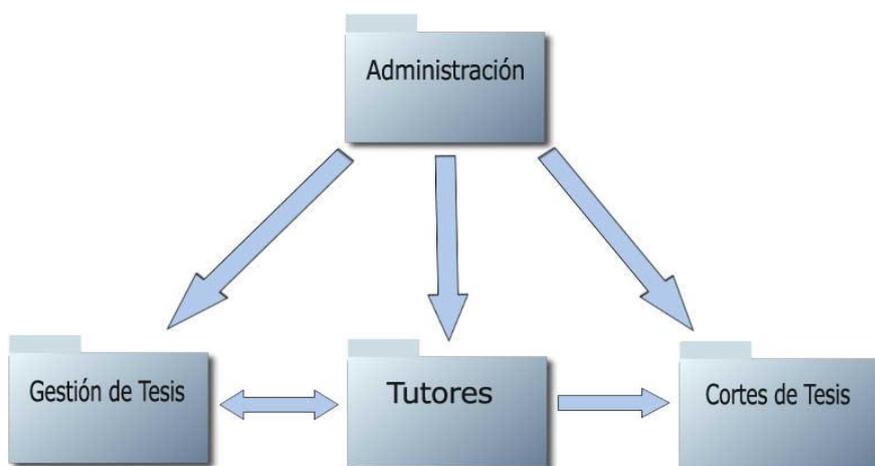


Figura 1 Diagrama de paquetes



2.3 Conclusiones del capítulo

Con el desarrollo de este capítulo se pudo comprender mejor la interacción del sistema con los usuarios. Se lograron definir de forma organizada todos los requisitos, tanto funcionales como no funcionales, los cuales sirvieron de guía en el proceso de implementación. También se logró precisar los posibles riesgos para el proyecto, así como la forma de mitigarlos, lo cual disminuye el efecto de los mismos, si ocurrieran.



CAPÍTULO 3: DESARROLLO DEL SISTEMA

3.1 Introducción

En este capítulo se presenta la fase de desarrollo de la metodología utilizada. En esta se describen cada unas de las tareas asociadas a cada historia de usuario y además se realiza el cronograma de producción en el cual se refleja cada una de las actividades a desarrollar por el equipo de trabajo.

3.2 Desarrollo

La fase de Desarrollo, es la segunda que define la metodología SXP. En la primera parte de esta fase se generan todos los documentos relacionados con la planificación de las iteraciones así como de las tareas a realizar durante la implementación.

3.2.1 Plan de releases

En este paso se define el plan de releases e iteraciones para realizar las entregas intermedias y la entrega final. Tiene como entrada la relación de historias de usuario definidas previamente. Para colocar una historia en cada iteración se tuvo en cuenta la prioridad de dicha historia. Como resultado de la priorización de historias se llegó a la siguiente planificación:



Tabla 6 Plan de realeses

Release	Descripción de la iteración	Orden de la HU a implementar	Duración total
1	En esta iteración se implementará una de las historias de usuario más importante del sistema ya que tienen que ver con la gestión de los temas de tesis	1	3
2	En esta iteración se implementará otra de las historias de usuario más importante del sistema ya que tiene que ver con la selección de los temas de tesis por los estudiantes.	2	3
3	En esta iteración se implementarán las historias de usuario que tienen que ver con la información de los cortes de tesis y por otra parte se implementará si un profesor desea ser tutor de una tesis.	3, 5	3
4	En esta iteración se implementará la historia de usuario que tiene que ver con la revisión de la tesis.	4	3

3.2.2 Cronograma de Producción

La plantilla de Cronograma de producción, es un documento de suma importancia, pues fue el que guió y controló las actividades que se realizaron



en cada iteración. En éste, no sólo se recogen las actividades planificadas, sino el tiempo que durarán y el rol responsable de desarrollarlas.

Para comprender cuales son los datos que se recogerán en esta plantilla, se muestra la misma gráficamente.

Tabla 7 Cronograma de producción

No	Hito	Descripción	Inicio	Fin	% ejec.	Ejecutor
1	Entrevista con profesores de las carreras de Eléctrica, Minas y Mecánica	Se desarrolló esta entrevista con el objetivo de conocer como se lleva a cabo el proceso de Trabajo de Diploma en estas carreras.	20/01/2010	20/01/2010	100%	Analista
2	Estudio de las herramientas a utilizar en el desarrollo de la aplicación	Esta actividad se realizó para poder conocer con mayor profundidad las herramientas a utilizar en la implementación del software.	5/01/2010	27/03/2010	90 %	Desarrollador
3	Diseño de la Base de Datos	Se diseñaron las tablas que se necesitarían para el manejo de la información.	17/02/2010	25/02/2010	100%	Diseñador
4	Diseño	Se diseñó la	26/02/2010	5/03/2010	95%	Diseñador



		interfaz de usuario.	10	010		r
5	Implementación	Se implementó la lógica del negocio.	1/02/2010	26/05/2010	97%	Desarrollador
6	Pruebas	Se realizaron pruebas a las diferentes historias de usuario.	28/05/2010	12/06/2010	90%	Desarrollador
7	Factibilidad	Se realizó un estudio de factibilidad.	27/05/2010	5/06/2010	100%	Analista

3.2.3 Tarjetas CRC

Las tarjetas CRC forman parte de las técnicas propuestas por algunos de los creadores de la metodología ágil XP (Ward Cunningham y Kent Beck), con el objetivo de obtener un diseño simple y que no incurra en la implementación de funcionalidades que no son necesarias. Esta técnica de modelado permite entender las características del sistema pensando en términos de objetos y clases.

Las tarjetas CRC permiten que el equipo completo contribuya en la tarea del diseño. Una tarjeta CRC representa un objeto. El nombre de la clase se coloca a modo de título en la tarjeta, los atributos y las responsabilidades más significativas se colocan a la izquierda, y las clases que están implicadas en cada responsabilidad a la derecha, en la misma línea que su requerimiento correspondiente.

Debido a la gran facilidad de uso y entendimiento, que propician dichas tarjetas, el equipo de desarrollo del presente trabajo, decidió utilizarlas para diseñar el sistema que se desea desarrollar.



Tabla 8 Tarjeta CRC <CTemasTesis>

Nombre de la clase: CTemasTesis	
Tipo de la clase: controladora	
Responsabilidades:	Colaboradores:
<ol style="list-style-type: none">1. Insertar un tema de tesis.2. Mostrar temas de tesis insertados.3. Aprobar temas de tesis.4. Mostrar temas de tesis aprobados.5. Aceptar temas de tesis propuestos.6. Mostrar temas de tesis escogidos.	

Tabla 9 Tarjeta CRC <CCorteTesis>

Nombre de la clase: CCorteTesis	
Tipo de la clase: controladora	
Responsabilidades:	Colaboradores:
<ol style="list-style-type: none">1. Insertar corte de tesis.2. Mostrar información de corte de tesis.	

Tabla 10 Tarjeta CRC <CTutorear>

Nombre de la clase: CTutorear
Tipo de la clase: controladora



Responsabilidades:	Colaboradores:
<ol style="list-style-type: none">1. Tutorar un tema de tesis.2. Mostrar temas de tesis con tutor.	

3.2.4 Tareas por Historias de Usuario

La plantilla de Tarea de ingeniería, posee una gran importancia, pues permite definir cada una de las actividades que estarán asociadas a las historias de usuario y que permitirán su implementación. También posibilita conocer que programador está asignado a cada tarea, así como el tiempo que se necesita para su realización, lo que facilita la estimación del tiempo que se llevará cada historia de usuario en implementarse, de acuerdo a su complejidad.

Para comprender cuales son los datos que se recogerán en esta plantilla, se muestra la misma gráficamente.

Tabla 11 Descripción de la tarea de ingeniería < Aprobar tema de tesis >

Tarea de Ingeniería	
Número Tarea: 01	Número Historia de Usuario: 1
Nombre Tarea: Aprobar tema de tesis.	
Tipo de Tarea : Desarrollo	Puntos Estimados: 1
Fecha Inicio:	Fecha Fin:
Programador Responsable: Viviana Nicot Hernández.	
Descripción: Un tema de tesis antes de ser propuesto en el sitio para que un estudiante lo realice debe ser aprobado antes por el jefe de la carrera o el jefe del departamento.	

Para ver las demás Plantillas de Tareas de ingeniería, ir al Expediente de Proyecto: 07- Tarea de ingeniería.



3.3 Conclusiones del capítulo

En este capítulo se confeccionaron las tarjetas CRC. Además se dividieron en tareas de ingeniería cada una de las historias de usuario para poder lograr la completa implementación de cada una en la fecha acordada con el cliente, lo cual permitió conocer el grado de complejidad de estas teniendo en cuenta la cantidad de tareas asociadas. A cada tarea de ingeniería se le asignó un tiempo de desarrollo que se cumplió de manera eficiente garantizando así el objetivo principal de su confección.



CAPÍTULO 4: PRUEBAS

4.1 Introducción

En el presente capítulo se muestran las pruebas de aceptación elaboradas por el cliente para demostrar que la aplicación funcione de forma correcta. Estas pruebas fueron llevadas a cabo antes de cada entrega que se realizó durante todo el desarrollo del proyecto.

4.2 Pruebas

Una de las mejores características de la metodología XP es el proceso de pruebas. Esta metodología propone probar tanto como sea posible. Esto permite aumentar la calidad de los sistemas reduciendo el número de errores no detectados y disminuyendo el tiempo transcurrido entre la aparición de un error y su detección. También permite aumentar la seguridad de evitar efectos no deseados a la hora de realizar modificaciones y refactorizaciones. XP propone la realización de pruebas unitarias, encargadas de verificar el código y diseñadas por los programadores, y pruebas de aceptación o pruebas funcionales destinadas a evaluar si al final de una iteración se consiguió la funcionalidad requerida por el cliente.

4.2.1 Pruebas de aceptación

En esta plantilla se escriben las pruebas ejecutadas según la historia de usuario seleccionada para realizar la comprobación y validar las



funcionalidades del sistema, y de esta forma saber si está apto para ser liberado.

A continuación se muestra un Caso de Prueba de Aceptación, el cual define su evaluación final.

Tabla 12 Caso de prueba de aceptación de la historia de usuario <Gestionar temas de tesis>

Caso de Prueba de Aceptación	
Código Caso de Prueba: SP-1-1	Nombre Historia de Usuario: Gestionar temas de tesis
Nombre de la persona que realiza la prueba: Viviana Nicot Hernández.	
Descripción de la Prueba: Se llenan los campos necesarios para poder insertar un tema de tesis	
Condiciones de Ejecución: Para poder insertar un tema de tesis el usuario debe especificar el título del tema de tesis, el resumen y el tipo de tesis que es esta.	
Entrada / Pasos de ejecución: el usuario inserta todos los datos del tema de tesis.	
Resultado Esperado: El sistema debe procesar toda la información sin mostrar un error.	
Evaluación de la Prueba: satisfactoria.	

Para ver las demás Plantillas Caso de prueba de aceptación, ir al Expediente de Proyecto: 09- Plantillas Caso de prueba de aceptación.

4.3 Conclusiones del capítulo

Con la realización de las pruebas de aceptación se asegura de que las funciones implementadas cumplan su objetivo satisfactoriamente, probando individualmente cada historia de usuario y asignándole la evaluación correspondiente. Todas las pruebas que se realizaron fueron positivas.



CAPÍTULO 5: ESTUDIO DE FACTIBILIDAD

5.1 Introducción

Para estudiar la factibilidad de este proyecto se utilizará la **Metodología Costo-Efectividad (Beneficio)**, la cual plantea que la conveniencia de la ejecución de un proyecto se determina por la observación conjunta de dos factores:

- El costo, que involucra la implementación de la solución informática, adquisición y puesta en marcha del sistema hardware/software y los costos de operación asociados
- La efectividad, que se entiende como la capacidad del proyecto para satisfacer la necesidad, solucionar el problema o lograr el objetivo para el cual se ideó, es decir, un proyecto será más o menos efectivo con relación al mayor o menor cumplimiento que alcance en la finalidad para la cual fue ideado (costo por unidad de cumplimiento del objetivo).

Esta tiene como objetivo fundamental proporcionar una medida de los costos que incurren en la realización de un proyecto informático, y a su vez comparar dichos costos previstos con los beneficios esperados de la realización de dicho proyecto.

5.2 Efectos Económicos

- Efectos directos.
- Efectos indirectos.
- Efectos externos.



- Intangibles.

5.2.1 Efectos directos

Efectos positivos

- Ahorro de tiempo para cumplir con las tareas a desarrollar en el proceso.
- Facilita la información necesaria a los estudiantes y profesores de los distintos departamentos implicados en el proceso.

Efectos negativos

- Para usar la aplicación es indispensable el uso de un ordenador, aparejado a los gastos que este trae de consumo de corriente eléctrica y mantenimiento.

5.2.2 Efectos indirectos

Los efectos económicos observados que pudiera repercutir sobre otros mercados no son perceptibles, aunque este proyecto no está construido con la finalidad de comercializarse.

5.2.3 Efectos externos

Se contará con una herramienta adecuada que permitirá a los usuarios gestionar y visualizar la información correspondiente al proceso de Trabajos de Diploma en el ISMM.

5.2.4 Intangibles

En la estimación financiera siempre hay elementos visibles por un grupo como perjudiciales o beneficiosos, pero al momento de darle valor en unidades monetarias esto resulta difícil o prácticamente imposible. A fin de medir con precisión los efectos, deberán considerarse dos situaciones:



Situación sin Proyecto

Cada estudiante al arribar al 5to año de la carrera busca proposiciones de temas de tesis y escoge el que desee pero lo tiene que hacer saber de forma personal. Luego que ya tiene su tema de tesis tiene que ver al tutor correspondiente con el tema para recibir orientaciones sobre lo que debe hacer, programar encuentros con éste y para que realice revisiones a su trabajo de tesis.

En este proceso se realizan varias evaluaciones para saber como va el desarrollo de la investigación, mas bien es un Proceso de Defensa de Trabajos de Diplomas de los cuales se da a conocer: fecha, distribución de los tribunales, el tipo de evaluación que se va a realizar (1er Corte, 2do Corte, etc.) y los objetivos a evaluar.

Situación con Proyecto

Mediante este sistema es posible gestionar y controlar de manera ágil y segura toda la información referente al proceso de Trabajos de Diploma en el ISMM. Además éste le da la posibilidad, tanto a un estudiante como a un profesor, de publicar algún tema de tesis luego de haber sido aprobado por el jefe de la carrera. También ofrece la ventaja a los profesores de poder ser tutor de alguna tesis propuesta en el sitio y en el caso de que ya tenga diplomante podrá realizar revisiones desde el sistema al Trabajo de Diploma. Esta aplicación posibilita que se visualice la información correspondiente a los cortes de tesis lo que permite que el estudiante se mantenga informado.

5.3 Beneficios Y Costos Intangibles en el proyecto

Costos:

- Resistencia al cambio.

Beneficios:

- Óptimo aprovechamiento del tiempo en la búsqueda de temas de tesis.
- Óptimo aprovechamiento del tiempo a la hora de buscar información.



- La tecnología utilizada es totalmente libre y gratuita por lo que no se incurre en gastos referentes al pago de licencias.
- Mejora en la calidad y visibilidad de la información.
- Mayor comodidad para todos los involucrados en el proceso.

5.4 Ficha de costo

Para determinar el costo económico del proyecto se utilizará el procedimiento para elaborar una Ficha de Costo de un Producto Informático [Dra. Ana María Gracia Pérez, UCLV]. Para la elaboración de la ficha se consideran los siguientes elementos de costo, desglosados en moneda libremente convertible y moneda nacional.

Costos en Moneda Libremente Convertible:

- **Costos Directos.**

1. Compra de equipos de cómputo: No procede.
2. Alquiler de equipos de cómputo: No procede.
3. Compra de licencia de Software: No procede.
4. Depreciación de equipos: \$ 60.78.
5. Materiales directos: No procede.

Total: \$ 60.78 CUC

- **Costos Indirectos.**

1. Formación del personal que elabora el proyecto: No procede.
2. Gastos en llamadas telefónicas: No procede.
3. Gastos para el mantenimiento del centro: No procede.
4. Know How: No procede.
5. Gastos en representación: No procede.

Total: \$0.00.

- **Gastos de distribución y venta.**

1. Participación en ferias o exposiciones: No procede.



2. Gastos en transportación: No procede.
3. Compra de materiales de propagandas: No procede.

Total: \$0.00.

Costos en Moneda Nacional:

• **Costos Directos.**

1. Salario del personal que laborará en el proyecto: \$100.00
2. El 5% del total de gastos por salarios se dedica a la seguridad social: No procede.
3. El 0.09% de salario total, por concepto de vacaciones a acumular: No procede
4. Gasto por consumo de energía eléctrica: \$ 5.94.
5. Gastos en llamadas telefónicas: No procede.
6. Gastos administrativos: No procede.

• **Costos Indirectos.**

1. Know How: \$ 108,75.

Total: \$ 214,69 MN

Como se hizo referencia anteriormente, la técnica seleccionada para evaluar la factibilidad del proyecto es la Metodología Costo-Efectividad. Dentro de esta metodología la técnica de punto de equilibrio aplicable a proyectos donde los beneficios tangibles no son evidentes, el análisis se basa exclusivamente en los costos. Para esta técnica es imprescindible definir una variable discreta que haga variar los costos. Teniendo en cuenta que el costo para este proyecto es despreciable, se tomará como costo el tiempo empleado por un estudiante para buscar un tema de tesis y el tiempo que emplea éste para contactar a su tutor y entregarle el documento de tesis para que se lo revise.

Para ello se ha dividido el proceso de la siguiente manera:



Valores de la variable (Solución manual)

1. El estudiante busca proposiciones de temas de tesis. (120 min.)
2. El estudiante contacta a su tutor. (30 min.)
3. El estudiante le entrega al tutor el documento de tesis a revisar. (10 min.)

Valor de la variable (Solución mediante el sistema)

1. El estudiante visualiza en el sistema las propuestas de temas de tesis. (1 min.)
2. El estudiante publica en el sistema el documento de tesis a revisar por el tutor. (1 min.)

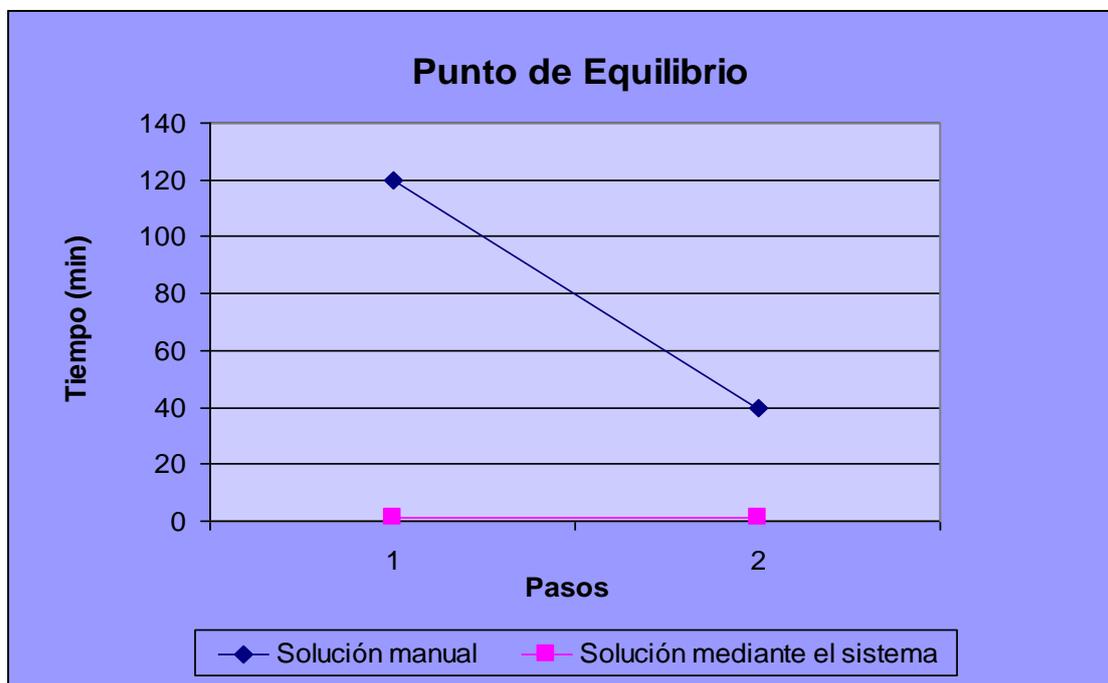


Figura 2 Gráfico de punto de equilibrio de soluciones

Para lograr una mejor comprensión del gráfico mostrado se debe explicar que algunas de las soluciones fueron concatenadas para lograr una semejanza con la solución mediante el sistema, de forma tal que:



Tabla 13 Explicación del gráfico de punto de equilibrio de soluciones

	Solución manual	Solución mediante el sistema
Paso 1	- El estudiante busca proposiciones de temas de tesis.	- El estudiante visualiza en el sistema las propuestas de temas de tesis.
Paso 2	- El estudiante contacta a su tutor. - El estudiante le entrega al tutor el documento de tesis a revisar.	- El estudiante publica en el sistema el documento de tesis a revisar por el tutor.

5.5 Conclusiones del capítulo

En este capítulo se realizó el estudio de factibilidad del proyecto mediante la Metodología Costo-Efectividad (Beneficio). En este se analizaron los efectos económicos, los beneficios y costos intangibles, y además se calculó el costo de ejecución del proyecto mediante la ficha de costo, la cual arrojó como resultado \$ 60.78 CUC y \$ 214.69 MN demostrándose la factibilidad del proyecto.



CONCLUSIONES GENERALES

Una vez concluida la investigación e implementación del sistema, se dieron cumplimiento a los objetivos planteados, obteniéndose los siguientes resultados:

- ❖ El estudio de cómo se lleva a cabo el proceso de Trabajo de Diploma en el ISMM permitió tener una visión más amplia del mismo, resultando más fácil su comprensión para el desarrollo del sistema.
- ❖ La realización de un estudio de las herramientas y tecnologías utilizadas permitió avanzar más rápido en la implementación de la aplicación.
- ❖ Se realizó todo el proceso de desarrollo del software siguiendo las fases de la metodología SXP, en las que quedaron plasmadas las tarjetas CRC, se desarrollaron las tareas correspondientes para dar solución a las historias de usuario y se realizaron las pruebas de aceptación las cuales demostraron la conformidad del cliente.
- ❖ Se desarrolló una aplicación Web dinámica que permitirá gestionar la información referente al proceso de Trabajo de Diploma en el ISMM.



RECOMENDACIONES

- ❖ Se propone la aplicación del software en el desarrollo de su trabajo como una de las acciones para validar el mismo y proponer su implementación en otras entidades.
- ❖ Implementar e incorporar las historias de usuarios que no se desarrollaron al Sistema de Pregrado para que ayude a ganar en eficiencia a la solución propuesta.



BIBLIOGRAFÍA

COLLAZO VÁZQUEZ, D.; O., CHANG PEREIRA. *El uso de las TIC en las Sedes Universitarias Municipales en Cuba y su influencia en el estrés de los estudiantes*. [en línea]. [Consultado: 2010-03-20]. Disponible en: <http://www.monografias.com/>

Desarrollo ágil de software. [en línea]. [Consultado: 2010-03-19]. Disponible en: http://es.wikipedia.org/wiki/Desarrollo_%C3%A1gil_de_software

GARCÍA MONTES, M.; O. GONZÁLEZ RICARDO. *Aplicación de las TIC en la Educación Superior*. [en línea]. [Consultado: 2010-04-8]. Disponible en: <http://www.monografias.com/>

LARÍN FONSECA, Y. *Sistema para el Control de recursos informáticos del Ministerio de Economía y Planificación (SISCRIN): módulo "Control de recursos informáticos"*. Yatsely Acea Choy; Karina Panucia Trabajo de Diploma. Instituto Superior Politécnico "José Antonio Echeverría", 2008. 84 hojas.

LETELIER, P.; M^a C. PENADÉS. *Metodologías ágiles para el desarrollo de software: eXtreme Programming (XP)*. [en línea]. [Consultado: 2010-04-8]. <http://www.willydev.net/descargas/masyxp.pdf>

MARTÍN W., F.; E. BALBIS COLL. *El trabajo investigativo estudiantil en la carrera de Mecánica de la Universidad de Cienfuegos*. [en línea]. [Consultado: 2010-04-8]. <http://www.bibliociencias.cu/gsd/collect/revistas/index/assoc/HASHb595/a4ce010c.dir/doc.pdf>



La Mejor Metodología 'Liviana' de Desarrollo de Software: eXtreme Programming. [en línea]. [Consultado: 2010-05-20]. Disponible en: <http://brconsulting.info/portal/articulos/metodologias-de-desarrollo/>

PEREZ ROLDÁN, D. *Sistema de Clonación y Distribución de Imágenes de Sistemas Operativos.* Abel Meneses Abad. Trabajo de Diploma. Universidad de las Ciencias Informáticas, 2008. 123 hojas.

Procesos de software: Metodologías Ágiles de Desarrollo de Software (Primera parte). [en línea]. [Consultado: 2010-02-15]. Disponible en: <http://www.qualitrain.com.mx/Metodologias-Agiles-de-Desarrollo-de-Software-Primera-Parte.html>

Programación extrema. [en línea]. [Consultado: 2010-02-8]. Disponible en: http://es.wikipedia.org/wiki/Programaci%C3%B3n_extrema

RODRÍGUEZ VILLAR, M. *Introducción de procedimientos ágiles en la producción de software en la facultad 7 de la Universidad de las Ciencias Informáticas.* Maidely Calderón Montero. Trabajo de Diploma. Universidad de las Ciencias Informáticas, 2007. 110 hojas.

Tesis - Wikipedia, la enciclopedia libre. (2010c). [en línea]. [Consultado: 2010-01-11]. Disponible en: <http://es.wikipedia.org/wiki/Tesis>

VILLEGAS, A.; E.A. PLAZA MARÍN; *Desarrollo de software bajo metodologías ágiles (Agile Methods XP) en la práctica.* [en línea]. [Consultado: 2010-02-15]. Disponible en: <http://www.monografias.com/>

WELICKI L.E. *Implementando Extreme Programming en la plataforma .NET.* [en línea]. [Consultado: 2010-02-23]. Disponible en:



<http://www26.brinkster.com/lwelicki/articles/Implementando%20Extreme%20Programming%20en%20la%20plataforma%20NET.pdf>



GLOSARIO DE TÉRMINOS

API: Una API (del inglés Application Programming Interface) es el conjunto de funciones y que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción.

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

Metodología Ágil: Constituyen un nuevo enfoque en el desarrollo de software, mejor aceptado por los desarrolladores de proyectos que las metodologías convencionales debido a la simplicidad de sus reglas y prácticas, su orientación a equipos de desarrollo de pequeño tamaño, su flexibilidad ante los cambios y su ideología de colaboración.

Metodologías de Desarrollo: Se define como un conjunto de filosofías, etapas, procedimientos, reglas, técnicas, herramientas, documentación y aspectos de formación para los desarrolladores de sistemas de información.

Open source: Código abierto o código libre. Software que distribuye de forma libre su código fuente, de forma que los desarrolladores pueden hacer variaciones, mejoras o reutilizarlo en otras aplicaciones. También conocido como free software.

PHP: Sus siglas, en español significan Procesador de Hipertexto, es un lenguaje de programación especialmente pensado para el desarrollo de aplicaciones en entorno Web.



Trabajo de Diploma: es una afirmación cuya veracidad ha sido argumentada, demostrada o justificada de alguna manera. Generalmente enuncia una proposición científica, un axioma o un hecho demostrable.

XML: Extensible Markup Language («lenguaje de marcas extensible»), es un metalenguaje extensible de etiquetas desarrollado por el World Wide Web Consortium (W3C). Permite la compatibilidad entre sistemas para compartir la información de una manera segura, fiable y fácil.



ANEXO 1

Modelo de datos

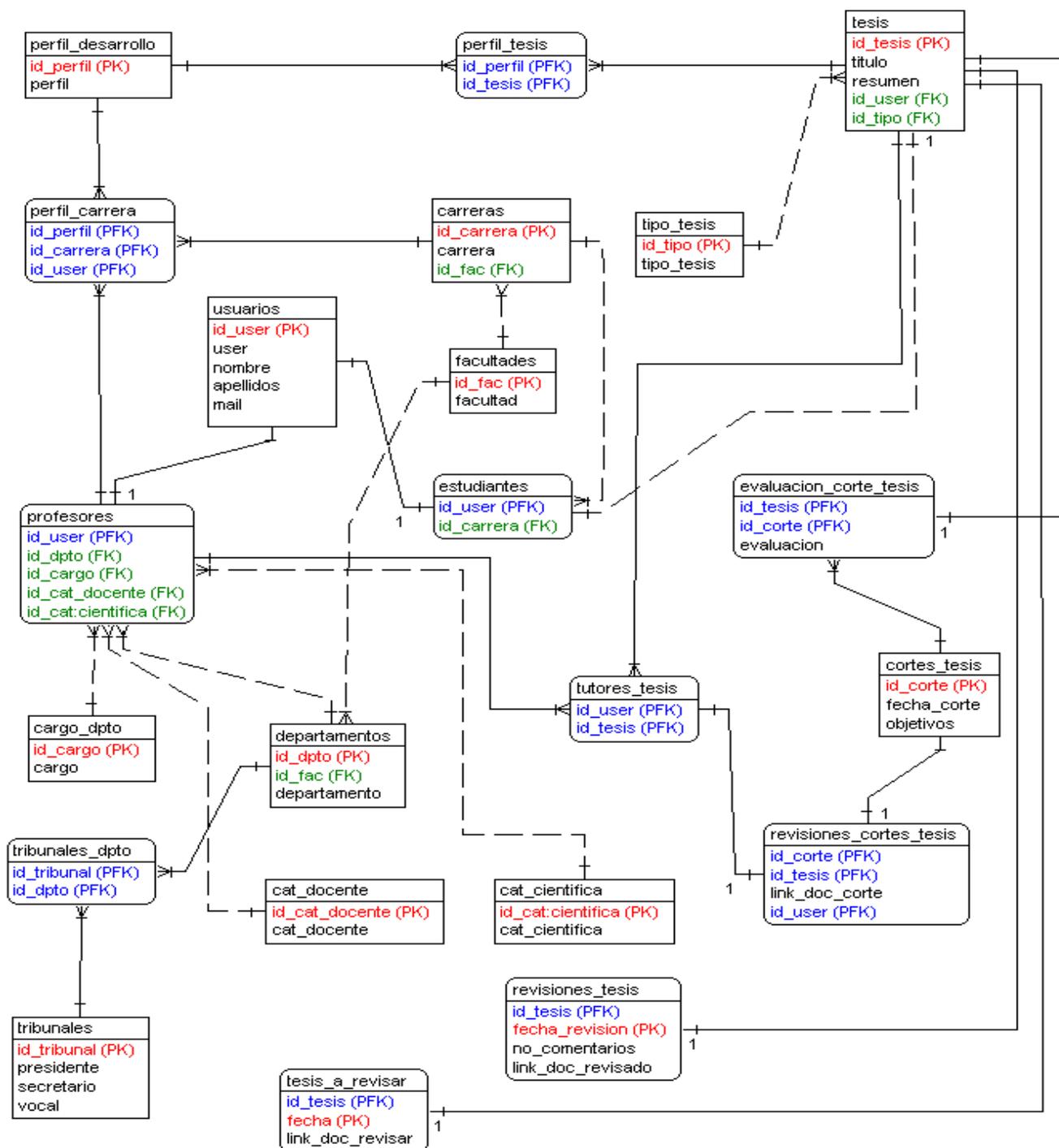


Figura 3 Modelo de datos