



INSTITUTO SUPERIOR MINERO METALÚRGICO

“Dr. Antonio Núñez Jiménez”.

Facultad de Metalurgia - Electromecánica

Moa, Holguín

Trabajo de Diploma

Para Optar por el Título de

Ingeniero Informático

“Gestión de planillas de usuarios para la solicitud de servicios informáticos en la red de computadoras en el ISMMM.”

Autor: Yasmany Chávez Cruz.

Tutor: Ing. Roilber Lambert Sánchez.

Moa, Cuba

Junio, 2011

Pensamiento.

“El conocimiento solo los necios lo evitan, conocer debería ser el único gusto que todo ser humano tenga en común.”

Yasmany Chávez Cruz

Agradecimientos.

De veras que existen muchas personas que me han ayudado en todo el transcurso de mi formación como persona y profesional y seria poco amable de mi parte no mencionarlo.

Ante todo un gran agradecimiento a mi tutor Roilber Lambert Sánchez que su ayuda me sirvió de gran enseñanza, ya que trabajar junto a él me dio la gran oportunidad de aprender muchas cosas de mi carrera que verdaderamente antes no sabía.

A mis compañeros y amigos que estuvieron conmigo estos cinco largos años de universidad que de una forma u otra aprendí algo de cada uno de ellos: Joandris, Laffita, Padilla, Humberto, Henry, Cala, Yanky, Eduardo, Yohandy y demás compañeros del grupo.

A mis padres, a mi suegro, a mi mujer y todos los demás que de una forma u otra contribuyeron a mi formación como persona y profesional.

A todos, gracias...

Yasmany Chávez Cruz.

Dedicatoria.

Después de estos largos años de estudios y sacrificios a quiénes más podría dedicarles este trabajo sino a las personas que han hecho que de una forma u otra yo esté aquí culminando esta gran etapa de mi vida, de que hoy sea quien soy y de que logre graduarme hoy de Ingeniero Informático; a las personas que por ellos existo hoy en el mundo, a mis Padres.

A otra persona que ha sido mi segunda madre y que yo se que ella lo siente así porque me lo ha demostrado en todo el transcurso de mi vida, a mi Abuela Materna.

A la persona que me ha sabido comprender, depositar en mí toda su confianza, estar conmigo en los buenos y malos momentos compartiendo cada uno de los sentimientos de amor, cariño, amistad; a mi alma gemela:

mi esposa Diurvis.

A la persona por la que actualmente mi vida tiene un verdadero sentido, a la que vino al mundo para darle amor y felicidad a mi vida, a mi bella princesita, mi querida hija; Ivian Yasmín Chávez Torres.

DECLARACIÓN DE AUTORÍA

Declaro que soy el único autor de este trabajo y autorizo al Instituto Superior Minero Metalúrgico de Moa “Dr. Antonio Núñez Jiménez” para que haga el uso que estime pertinente con este trabajo.

Para que así conste firmo la presente a los 26 días del mes de junio del 2011.

Yasmany Chávez Cruz

Nombre completo del primer autor

Roilber Lambert Sánchez

Nombre Completo del tutor

OPINIÓN DEL USUARIO DEL TRABAJO DE DIPLOMA

El Trabajo de Diploma, titulado Gestión de planillas de usuarios para la solicitud de servicios informáticos en la red de computadoras en el ISMMM, fue realizado en el 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 informe técnico le reportan a esta entidad los beneficios siguientes (cuantificar):

Como resultado de la implantación de este trabajo se reporta un efecto económico que asciende a <valor> en MN y/o <valor> CUC. (Este valor debe ser REAL, no indica lo que se reportará, sino lo que reporta a la entidad. Puede desglosarse por conceptos, tales como: cuánto cuesta un software análogo en el mercado internacional, valor de los materiales que se ahorran por la existencia del software, valor anual del (de los) salario(s) equivalente al tiempo que se ahorra por la existencia del software).

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

Nombre del representante de la entidad

Cargo

Firma

Cuño

OPINIÓN DEL TUTOR DEL TRABAJO DE DIPLOMA

RESUMEN

En el presente trabajo se define en detalle el proceso de Gestión y Aprobación de Solicitudes de Servicios Informáticos en el ISMMM. Se realiza un estudio acerca de los diferentes sistemas existentes vinculados al campo de acción. Se presentan las herramientas a utilizar en el desarrollo del sistema propuesto y como metodología de desarrollo de software XP (Programación Extrema), abordando en detalles cada una de sus fases.

Se presentan las principales definiciones dentro de la implementación así como las interfaces gráficas diseñadas. Se describen además las pruebas realizadas y sus resultados, y se realiza un estudio de los esfuerzos requeridos para la construcción del sistema.

Se desarrolló un sistema capaz de gestionar el proceso de aprobación de solicitudes de servicios informáticos en el centro cumpliendo con esto el objetivo principal del trabajo.

ABSTRACT

In this paper defines in detail the process of Application and Approval Management Information Services in ISMMM. It's carried out a study about the different existing systems linked to the field. It presents the tools used in developing the proposed system, as well as the software development methodology XP (Extreme Programming), detailing each of its phases.

The main definitions are present in the implementation as well as graphical interfaces designed. It also describes the tests and their results, and a study of the efforts required to develop the system.

We developed a system capable of managing the approval process of applications and computing services at the institute in compliance with the main objective of this work.

Tabla de Contenido.

Pensamiento.....	I
Agradecimientos.....	II
Dedicatoria.....	III
DECLARACIÓN DE AUTORÍA	IV
OPINIÓN DEL USUARIO DEL TRABAJO DE DIPLOMA	V
OPINIÓN DEL TUTOR DEL TRABAJO DE DIPLOMA	VI
RESUMEN	VII
INTRODUCCIÓN	1
CAPÍTULO I – FUNDAMENTACIÓN TEÓRICA.....	6
1.1 Introducción.....	6
1.2 Estado del arte.....	6
1.2.1 Sistemas automatizados existentes vinculados al campo de acción.....	6
1.3 Tendencias y tecnologías actuales para el desarrollo de aplicaciones.....	8
1.3.1 Lenguaje de programación utilizado.....	8
1.3.2 Framework de Desarrollo utilizado.....	10
1.3.3 Sistema Gestor de Base de Datos utilizado (SGBD).....	11
1.3.3 Servidor Web utilizado.....	13
1.3.4 Herramientas utilizadas.....	13
1.3.4.1 Paquete preconfigurado XAMPP.....	13
1.3.4.2 Herramienta ER/Studio.....	14
1.3.4.3 PhpDesigner.....	14
1.4 Metodología para el desarrollo de sistemas informáticos.....	15
1.4.1 Metodología utilizada XP (Extreme Programming)	16
1.5 Conclusiones del capítulo.....	18
CAPÍTULO II - PLANEACIÓN Y DISEÑO	19
2.1 Introducción	19
2.2 Funcionalidades generales	19
2.2.1 Personal relacionado con el sistema.....	20
2.2.2 Lista de reserva.....	20
2.2.3 Historias de usuario.....	20
2.3 Planificación de entregas	22
2.3.1 Estimación de esfuerzo por historias de usuario	23
2.3.2 Planificación de iteraciones.....	24
2.3.3 Plan de duración de las iteraciones	26
2.4 Clases, responsabilidades y colaboradores.....	27

2.5 Conclusiones	29
CAPITULO III – DESARROLLO Y PRUEBAS	30
3.1 Introducción	30
3.2 Modelo de datos	30
3.3 Desarrollo de las iteraciones	32
3.3.1 Tareas por historias de usuario	32
3.4 Pruebas	35
3.4.1 Desarrollo dirigido por pruebas	36
3.4.2 Pruebas de Aceptación.....	36
3.5 Conclusiones	38
CAPITULO IV - ESTUDIO DE FACTIBILIDAD	39
4.1 Introducción	39
4.2 Métodos para la estimación del esfuerzo	39
4.3 Estudio de factibilidad	40
4.4 Análisis de sostenibilidad	47
4.5 Beneficios tangibles e intangibles	48
CONCLUSIONES GENERALES.....	50
RECOMENDACIONES	51
REFERENCIAS BIBLIOGRÁFICAS	52
GLOSARIO DE TERMINOS.....	54
ANEXO 1: Planillas de solicitud de Servicios	55
ANEXO 2: Continuación de la lista de reserva del producto	57
ANEXO 3: HISTORIAS DE USUARIO	59
ANEXO 4: TARJETAS CRC.....	63
ANEXO 6: TARJETAS DE TAREAS.....	67
ANEXO 7: PRUEBAS DE ACEPTACIÓN	80
ANEXO 8: INTERFASES DE USUARIO.....	86



INTRODUCCIÓN

El mundo de hoy se caracteriza por el desarrollo acelerado de las tecnologías. Ningún centro, empresa o negocio podrá tener éxito si no es capaz de adaptarse a los diferentes cambios aprovechando los beneficios de esta nueva era, en la que la informática rige cada vez más el proceso de desarrollo y evolución del ser humano, donde esta le permite automatizar cada uno de los procesos donde el hombre se ve involucrado, simplificándole con esto disímiles gastos económicos y sociales.

La automatización solo es viable, si al evaluar los beneficios económicos y sociales de las mejoras que se podrían obtener al automatizar, son mayores a los costos de operación y mantenimiento del sistema, sin embargo, no hay dudas de que la automatización de un proceso frente a la ejecución manual del mismo, ofrece ventajas de orden económico, social, y tecnológico; esto se debe a que se asegura una mejora en la calidad del trabajo por parte del que lo realiza y en el desarrollo del proceso, la que dependerá de la eficiencia del sistema implementado.(Aguilera, 2010)

En el ISMMM el centro de red de computadoras del mismo presta a sus usuarios una serie de servicios informáticos los cuales para su asignación deben pasar por un estricto proceso de aprobación. Cada solicitud debe estar aprobada por cada uno de los facultativos pertinentes; para poder realizar la asignación de los servicios solicitados.

Actualmente este proceso de petición de servicios es efectuado de forma manual y a veces se convierte en un proceso lento y poco confiable; es decir, en ocasiones se realiza una solicitud y esta se queda en la mitad del proceso y el usuario que realizó dicha solicitud no recibe respuesta de su petición, lo que ocasiona inconformidad para el solicitante. Además todos los gastos de papel y otros materiales de oficina



que traen consigo este proceso y el incremento de las posibilidades de que se produzcan errores en la manipulación de la información del mismo, motivo por el cual se ha presentado la necesidad de realizar la automatización de ciertas actividades que son repetitivas y que son factibles de informatización.

El sistema estará al alcance de los administradores de departamentos, decanos, Vice Rector de Investigación y Postgrado y administrativo de red, que son los que van a gestionar el proceso y, además, estará disponible para que los usuarios puedan visualizar si se le aprobó su solicitud individual.

Todo lo antes expuesto permite revelar la poca eficiencia del proceso de gestión de solicitud de servicios informáticos en el Instituto Superior Minero Metalúrgico de Moa “Antonio Núñez Jiménez”, por lo cual se plantea como **Problema Científico:**

“La ausencia de un Sistema Informático que automatice el proceso de Gestión de Solicitud y Aprobación de Acceso a los Servicios Informáticos de la red de computadoras del ISMM”

El **Objeto de estudio** se define como el desarrollo de aplicaciones Web para la Gestión de Solicitud y Aprobación de Acceso a los Servicios Informáticos. Específicamente se investigará en el siguiente **campo de acción:** Desarrollo de una aplicación Web para la gestión y aprobación de solicitudes de servicios informáticos en el ISMMM utilizando la metodología XP.

El **Objetivo general** que se persigue es desarrollar una aplicación Web que permita gestionar el proceso de solicitud y aprobación de servicios informáticos del ISMM y su accesibilidad al mismo de los directivos que lo gestionan a través de la Web; contribuyendo a una mejora sostenida en el proceso de generación de beneficios en el ISMMM.



Objetivos específicos.

1. Realizar observaciones en el campo del objeto de estudio y entrevistas realizadas para definir bien el proceso.
2. Estudiar la metodología para la elaboración del software.
3. Realizar el análisis y diseño del software para su posterior implementación.
4. Desarrollar un software capaz de gestionar solicitudes de servicios informáticos en el ISMM.

La idea a defender es que: la creación de un sistema automatizado para la gestión de solicitudes de servicios informáticos, agilizará el desarrollo de esta actividad en el ISMMM y por consiguiente contribuirá a una asignación de servicios más eficiente en el centro.

Para resolver el problema planteado, se proponen las siguientes **Tareas**:

- ✚ Definición detallada del proceso a automatizar.
- ✚ Hacer un estudio de cómo sería viable la automatización del proceso.
- ✚ Entrevistas, búsquedas y recopilación de información concerniente al objeto de estudio.
- ✚ Recopilación de información sobre la Metodología XP
- ✚ Realizar la Ingeniería de Software siguiendo la metodología antes mencionada.
- ✚ Realizar trabajo de mesa detallado de las distintas interfases, sistema de variables, bases de datos, definir las principales funcionalidades, etc.
- ✚ Desarrollar la aplicación Web utilizando como lenguaje de programación base el PHP.
- ✚ Realización de pruebas previas para verificar la eficiencia del sistema.
- ✚ Determinar la Factibilidad y Sostenibilidad del sistema.



Para darle cumplimiento a estas tareas los **métodos de investigación** empleados fueron los **teóricos** y **empíricos**. Entre los **métodos empíricos** se puede citar la entrevista y el análisis de documentos para la recopilación de la información. Mediante el análisis de la documentación y del proceso se supo cual era la funcionalidad del mismo para su posterior gestión; la entrevista permitió conocer en detalles cuales eran las necesidades de los directivos para la informatización de la gestión de solicitudes de servicios informáticos y además para determinar los requerimientos funcionales que debe cumplir el sistema a desarrollar.

Los **métodos teóricos** utilizados fueron: el método histórico y lógico para la búsqueda de los antecedentes del software. El análisis y síntesis para la recopilación y el procesamiento de la información obtenida en los métodos empíricos y arribar a las conclusiones de la investigación; y el hipotético deductivo para la elaboración de la idea a defender y su verificación. Mediante la modelación se realizó el estudio de la gestión de solicitudes de servicios informáticos en el ISMMM.

Las etapas de la investigación son:

- El estudio de las distintas metodologías de desarrollo de software, de las herramientas, tecnologías y patrones arquitectónicos para hacer una selección de las mejores, y utilizarlos en la confección del software de acuerdo a sus características.
- La contextualización del problema, enmarcando los objetivos específicos requeridos por el cliente.
- Análisis y diseño del software con todas las especificaciones requeridas.
- Realización del estudio de factibilidad.

El presente trabajo consta de introducción, 4 capítulos, conclusiones, recomendaciones, bibliografía, glosario de términos y anexos:



En el capítulo 1, Fundamentación teórica, se analizan aspectos relacionados con: definiciones esenciales, se realiza un estudio acerca de los diferentes sistemas existentes vinculados al campo de acción, además se presenta la metodología y las herramientas a utilizar en el desarrollo del sistema propuesto.

En el capítulo 2, Planeación y Diseño, se hace uso de la metodología expuesta en el capítulo inicial para el desarrollo del proyecto, abordando en detalles cada una de sus fases.

El capítulo 3, Desarrollo y Pruebas, se presentan los principales métodos y definiciones dentro de la implementación de los flujos de trabajo. También se muestran las interfaces gráficas diseñadas para la interacción de los flujos de trabajo con los usuarios. Se describen además las pruebas realizadas y sus resultados.

En el capítulo 4, Estudio de Factibilidad y Sostenibilidad, se realiza un estudio de los esfuerzos requeridos para la construcción del sistema, y se valora la sostenibilidad del producto.

Para concluir se muestran las Conclusiones a las que se arribaron, las Recomendaciones propuestas, la Bibliografía empleada, Glosario de Términos y Anexos con la información necesaria sobre el trabajo.



CAPÍTULO I – FUNDAMENTACIÓN TEÓRICA.

1.1 Introducción.

Gracias al desarrollo de la tecnología y junto a ella la informática se ha logrado resolver disímiles problemas a través de la producción de diversos software. En este capítulo se realiza un esbozo acerca de cómo es el proceso de gestión de solicitud de servicios informáticos, así como sus definiciones fundamentales, además se exponen tanto la metodología a seguir en el desarrollo de la aplicación, como las herramientas a emplear en la misma.

1.2 Estado del arte.

El proceso de solicitud de servicios informáticos en el centro funciona de la siguiente manera:

En el ISMMM el centro de red de computadoras del mismo presta a sus usuarios una serie de servicios informáticos los cuales para su asignación deben pasar por un estricto proceso de aprobación: los usuarios realizan las solicitudes de servicios ([anexo 1](#)) desde sus respectivos departamentos a los decanos de sus facultades, estos aprueban o rechazan dichas solicitudes y las solicitudes aprobadas por los decanos son remitidas al Vicerrector de Investigación y Postgrado(V.R.I.P) y el mismo aprueba o rechaza dichas solicitudes y las aprobadas son entregadas al administrativo de red que es el encargado de asignar los servicios.

1.2.1 Sistemas automatizados existentes vinculados al campo de acción.

Se encontraron varios sistemas que gestionan solicitudes de servicios pero no todos gestionan solicitudes de servicios informáticos. Existe un sistema que presta servicios y algunos son informáticos como por ejemplo acceso al correo electrónico, lo que no se conoce si se sigue el mismo proceso para aceptar las solicitudes de servicios.

Sitio Web de la Facultad de Medicina de la Universidad de Valladolid.

Fig. 1 <Imagen principal del sitio Web de la Facultad de Medicina de la Universidad de Valladolid>

Este sistema es una aplicación Web que tiene varias funcionalidades, las cuales dentro de las que nos concierne en la investigación están, solicitar servicios y prestar servicios.

Sitio Web de la Universidad de Oviedo.

Existe otra aplicación Web en la Universidad de Oviedo (fig.2) que de cierta manera gestiona solicitudes de servicios informáticos para el personal que en ella radica; es como una especie de página informativa que le muestra a sus usuarios de cómo funciona el proceso de solicitud además de permitirles también la descarga de los formularios de solicitud de los diferentes servicios que presta.

Para acceder a los diversos servicios telemáticos que presta la Universidad de Oviedo es necesario presentar credenciales de acceso que identifiquen de forma unívoca a la persona que está accediendo al servicio.

Actualmente esas credenciales están formadas por un par usuario/contraseña. El par usuario/contraseña puede ser diferente dependiendo del servicio al que se esté tratando de acceder.

Citado en: (www.dgr.gub.uv/gestionusuarios.asp.htm)

Los diferentes usuarios que nos podemos encontrar son: Usuario Corporativo, Usuario de Acceso Remoto, Usuario de Gestión y Usuario de Aplicación; a continuación se muestra una imagen de la portada del sitio.



Fig. 2 <Imagen principal del sitio Web de la Universidad de Oviedo >

Después de definir el proceso y hacer un estudio de las diferentes aplicaciones vinculadas al campo de acción se arribaron a las siguientes conclusiones:

Los sistemas encontrados no siguen el mismo proceso de gestión de solicitudes de nuestro centro. Además de que no se prestan los mismos servicios y estos son implementados a la necesidad de esas universidades. Por lo que se plantea la necesidad de implementar un sistema capaz de gestionar el proceso de solicitud de servicios informáticos en el ISMMM de acuerdo a nuestras necesidades y siguiendo nuestro proceso de aprobación.

1.3 Tendencias y tecnologías actuales para el desarrollo de aplicaciones.

1.3.1 Lenguaje de programación utilizado.

Personal Home Page - (PHP) es el acrónimo de procesador hipertexto (Hipertexto Preprocessor). Es un lenguaje de programación del lado del servidor gratuito e independiente de plataforma, rápido, con una gran librería de funciones y mucha documentación. Es también un lenguaje interpretado y embebido en el HTML. Fue creado originalmente en 1994 por Rasmus Lerdorf, pero como PHP está desarrollado en política de código abierto, a lo largo de su historia ha tenido muchas contribuciones de otros desarrolladores.

PHP es un lenguaje de programación de estilo clásico, esto significa que es un lenguaje de programación con variables, sentencias condicionales, bucles, funciones, etc. No es un lenguaje de marcas como podría ser HTML, XML o WML.



A diferencia de Java o JavaScript que se ejecutan en el navegador, PHP se ejecuta en el servidor, por eso permite acceder a los recursos que tenga el servidor, como por ejemplo podría ser, una base de datos. El programa PHP es ejecutado en el servidor y el resultado enviado al navegador. El resultado es normalmente una página HTML pero igualmente podría ser una página WML.

PHP es la gran tendencia en el mundo de Internet. Últimamente se puede observar un ascenso imparable, puesto que cada día son muchas las páginas Web que lo utilizan para su funcionamiento, según las estadísticas, PHP se utiliza en más de 10 millones de páginas, y cada mes realiza un aumento del 15%. Como síntesis, PHP corre en 7 plataformas, funciona en 11 tipos de servidores, ofrece soporte sobre unas 20 bases de datos tales como MySQL, Postgres, Oracle, ODBC, DB2, Microsoft SQL Server, Firebird y SQLite; lo cual permite la creación de aplicaciones Web muy robustas, y contiene unas 40 extensiones estables sin contar las que se están experimentando, además de que:

- Es software libre, lo que implica menos costes y servidores más baratos que otras alternativas.
- Es muy rápido. Su integración con la base de datos MySQL y el servidor Apache, le permite constituirse como una de las alternativas más atractivas del mercado.
- Su sintaxis está inspirada en C, ligeramente modificada para adaptarlo al entorno en el que trabaja, de modo que si está familiarizado con esta sintaxis, resultará un poco mejor aprender PHP.
- Su librería estándar es realmente amplia, lo que permite reducir los llamados “costes ocultos”, uno de los principales defectos de ASP.
- PHP tiene una de las comunidades más grandes en Internet, esto permite encontrar fácilmente ayuda, documentación, artículos, noticias y otros recursos.
- Permite las técnicas de Programación Orientada a Objetos (POO).
- Posibilita crear los formularios para la Web.



- No requiere definición de tipos de variables ni manejo detallado del bajo nivel.(Aguilera, 2010)

1.3.2 Framework de Desarrollo utilizado.

CodeIgniter:

Es un poderoso Framework para PHP que facilita la escritura de código repetitivo, y a comparación de otros Frameworks cómo CakePHP, Symfony 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 interfase 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". CodeIgniter está basado en el patrón de desarrollo Modelo-Vista-Controlador. MVC es una aproximación al software que separa la lógica de la aplicación de la presentación. En la práctica, permite que sus páginas Web contengan mínima codificación ya que la presentación es separada del código PHP. Este framework tiene un enfoque bastante flexible del MVC, ya que los Modelos no son requeridos. Si no se necesita agregar separación, o se descubre que mantener los modelos requiera más complejidad de la que se quiere, puede ignorarse y construir la aplicación usando Controladores y Vista. También permite incorporar sus códigos existentes, o incluso desarrollar librerías de núcleo para el sistema, habilitándolo a trabajar en una forma que hace que tenga más sentido para el desarrollador.(Lozano)

El siguiente gráfico ilustra como fluyen los datos a través del sistema:

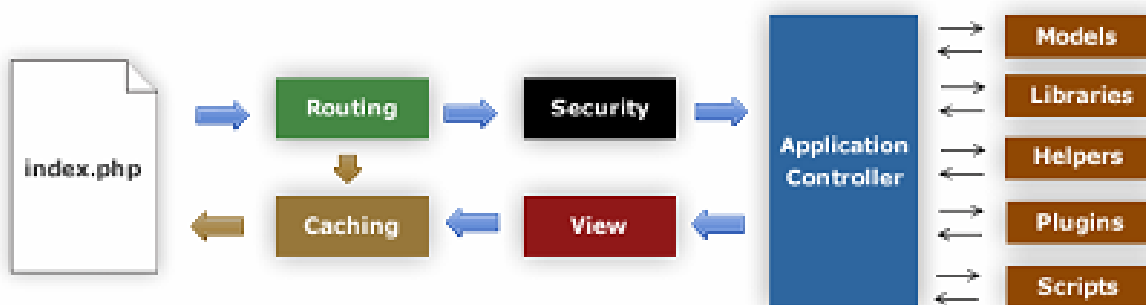


Figura 4: Flujo de la Aplicación con Codeigniter.

1. El **index.php** sirve como controlador frontal, inicializando los recursos básicos necesarios para correr Codeigniter.
2. El **Router** examina la petición HTTP para determinar que debe ser hecho con él.
3. Si un **archivo de caché** existe, es enviado directamente al explorador, sobrepasando el sistema de ejecución normal.
4. **Seguridad**. Antes que el controlador sea cargado, la petición HTTP y cualquier dato suministrado por el usuario es filtrado por seguridad.
5. El **controlador** carga los modelos, librerías, plugins, asistentes y cualquier otro recurso necesario para procesar la petición específica.
6. La **Vista** finalizada es presentada y enviada al explorador Web. Si el cacheo está habilitado, la vista es cacheada primero para que las peticiones subsecuentes puedan ser servidas.(Myer)

1.3.3 Sistema Gestor de Base de Datos utilizado (SGBD).

MySQL - es un sistema de gestión de base de datos relacional, multi-hilo y multiusuario, con más de seis millones de instalaciones. MySQL AB desarrolla MySQL como software libre en un esquema de licenciamiento dual. MySQL AB pertenece a Sun Microsystems desde enero de 2008. Por un lado lo ofrece bajo la GNU GPL, pero, empresas que quieran incorporarlo en productos privativos pueden



comprar a la empresa una licencia que les permita ese uso. Está desarrollado en su mayor parte en ANSIC.

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

Esto es lo que posibilita el esquema de licenciamiento anteriormente mencionado. Además de la venta de licencias privativas, la compañía ofrece soporte y servicios. Para sus operaciones contratan trabajadores alrededor del mundo que colaboran vía Internet. MySQL AB fue fundado por David Axmark, Allan Larsson, y Michael Widenius.

MySQL funciona sobre múltiples plataformas, incluyendo AIX, BSD, FreeBSD, HP-UX, GNU/Linux, Mac OS X, NetBSD, Novell Netware, OpenBSD, OS/2 Warp, QNX, SGI IRIX, Solaris, SunOS, SCO OpenServer, SCO UnixWare, Tru64, Windows 95, Windows 98, Windows NT, Windows 2000, Windows XP, Windows Vista y otras versiones de Windows.

Luego de analizadas las características y facilidades del SGBD presentado, y la de la herramienta a desarrollar se decide usar el MySQL como SGBD, por las siguientes ventajas además de las anteriormente mencionadas:

- No se necesitará de un manejo complejo de la información.
- El PHP maneja más fácil al MySQL que al SQL Server, debido a la gran cantidad de funciones que tiene explícitas.
- El MySQL es multiplataforma.
- El MySQL no tiene precio en el mercado, se adquiere libremente.

Citado en: (<http://es.wikipedia.org/wiki/MySQL>)



1.3.3 Servidor Web utilizado.

Apache - Es el servidor Web más utilizado en el mundo con un 57 % de cuota de mercado, frente al 20 % de Microsoft IIS y el 7 % de Netscape. Dentro de sus puntos fuertes se encuentran:

- Tiene interfaz con todos los sistemas de autenticación.
- Facilita la integración como "plug-ins" de los lenguajes de programación de páginas Web dinámicas más comunes.
- Tiene integración en estándar del protocolo de seguridad SSL. (más utilizado).
- Provee interfaz a todas las bases de datos.

1.3.4 Herramientas utilizadas.

1.3.4.1 Paquete preconfigurado XAMPP.

XAMPP es un paquete preconfigurado que permite instalar varios tipos de servidores en tu sistema con unos pocos clicks y en tan solo 5 minutos y sin la necesidad de conocer muchos de los servidores. Incluye el servidor Web Apache, los servidores de Bases de Datos MySQL y SQLite con sus correspondientes gestores phpMyAdmin y phpSQLiteAdmin respectivamente, el intérprete del lenguaje de programación PHP en su última versión, con algunos extras incluidos como el PEAR, el intérprete del lenguaje de programación Perl, servidores de FTP como ProFTP o FileZilla FTP Server, las estadísticas Webalizer, y OpenSSL y otros agregados.

Existen versiones para **GNU/Linux**, **Windows**, **MacOS X** y **Solaris**, cuyos paquetes difieren según la disponibilidad de los diversos programas en cada plataforma.(BLANCO CRIADO, 2008)



1.3.4.2 Herramienta ER/Studio.

Es una herramienta de modelado de datos fácil de usar y multinivel, para el diseño y construcción de bases de datos a nivel físico y lógico. Direcciona las necesidades diarias de los administradores de bases de datos, desarrolladores y arquitectos de datos que construyen y mantienen aplicaciones de bases de datos grandes y complejas.

ER/Studio está equipado para crear y manejar diseños de bases de datos funcionales y confiables. Ofrece fuertes capacidades de diseño lógico, sincronización bidireccional de los diseños físicos y lógicos, construcción automática de bases de datos, documentación y fácil creación de reportes.

ER/Studio ofrece las siguientes funcionalidades:

Capacidad fuerte en el diseño lógico.

Sincronización bidireccional de los diseños lógico y físico.

Construcción automática de Base de Datos.

Reingeniería inversa de Base de Datos.

Documentación basada en HTML.

Un Repositorio para el modelado.

Potencial de ER/Studio.

Diagramas.

En:(http://bureaudeprensa.com/es/view.php?bn=bureaudeprensa_software&key=1153755975)

1.3.4.3 PhpDesigner.

PhpDesigner 2008 v6.0.0 - Es un completo entorno de desarrollo y programación especialmente diseñado para los gurús de PHP, aunque también permite trabajar con comodidad en otros lenguajes de programación como HTML, XHTML, CSS y SQL.



Ofrece toda una serie de asistentes y diálogos integrados que facilitan en todo momento las tareas, además de acceso directo a librerías de código o scripts de uso habitual, utilidades diversas y toda suerte de herramientas, todo ello en una interfaz de diseño sencillo y elegante que se puede personalizar con nada menos que dieciocho temas distintos.

Cuenta con cliente de FTP y navegador de ficheros integrado, utilidades de corrección y autocompletado, búsqueda integrada en google y soporte para proyectos, además de usar un práctico esquema de color para la sintaxis del código fuente que facilita enormemente la programación. PhpDesigner soporta: PHP, HTML, XHTML, CSS, Java, Perl, JavaScript, VB, C# y SQL.

Citado en: (<http://www.intercambiosvirtuales.org/software/php-designer-2008-v6020-professional>)

1.4 Metodología para el desarrollo de sistemas informáticos.

El desarrollo de software no es una tarea fácil. Prueba de ello es que existen numerosas propuestas metodológicas que inciden en distintas dimensiones del proceso de desarrollo. Por una parte tenemos aquellas propuestas más tradicionales que se centran especialmente en el control del proceso, estableciendo rigurosamente las actividades involucradas, los artefactos que se deben producir, las herramientas y notaciones que se usarán. Estas propuestas han demostrado ser efectivas y necesarias en un gran número de proyectos, pero también han presentado problemas en muchos otros.

Sin embargo, el resultado final sería un proceso de desarrollo más complejo que puede incluso limitar la propia habilidad del equipo para llevar a cabo el proyecto. Otra aproximación es centrarse en otras dimensiones, como por ejemplo el factor humano o el producto software. Esta es la filosofía de las metodologías ágiles, las cuáles dan mayor valor al individuo, a la colaboración con el cliente y al desarrollo incremental del software.



Las metodologías ágiles están revolucionando la manera de producir software, y a la vez generando un amplio debate entre sus seguidores y quienes por escepticismo o convencimiento no las ven como alternativa para las metodologías tradicionales. (FERRER)

1.4.1 Metodología utilizada XP (Extreme Programming)

XP (Extreme Programming) es una metodología creada por Kent Beck, que se centra 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. Se basa en la realimentación continua entre el cliente y el equipo de desarrollo, la comunicación fluida entre todos los participantes, la simplicidad en las soluciones implementadas y el coraje para enfrentar los cambios. Se define como especialmente adecuada para proyectos con requisitos imprecisos y muy cambiantes, donde existe un alto riesgo técnico. Los principios y prácticas son de sentido común pero llevadas al extremo, de ahí proviene su nombre.

Características

- Su desarrollo es iterativo e incremental con pequeñas mejoras, unas tras otras.
- Se realizan pruebas unitarias continuas. Se aconseja escribir el código de la prueba antes de la codificación.
- Hay una frecuente interacción del equipo de programación con el cliente o usuario. Se recomienda que un representante del cliente trabaje junto al equipo de desarrollo.
- Se corrigen todos los errores antes de añadir una nueva funcionalidad y se hacen entregas frecuentes.
- Hay una refactorización del código, es decir, se reescriben ciertas partes del código para aumentar su legibilidad y mantenimiento pero sin modificar su comportamiento.
- Exige simplicidad en el código, siendo esta la mejor manera de que las cosas funcionen. Cuando todo funcione se podrán añadir funcionalidades si es necesario. La programación extrema apuesta que es más sencillo hacer algo simple y tener un



poco de trabajo extra para cambiarlo si se requiere, que realizar algo complicado y quizás nunca utilizarlo (Álvarez, y otros, 2007). XP no es un modelo de procesos ni un marco de trabajo, sino un conjunto de prácticas que se complementan unas a otras y que ofrecen una base sólida para un óptimo desempeño, alta productividad e inestimables beneficios y deben implementarse en un entorno de desarrollo cuya cultura se base en cinco valores:

1. Simplicidad: enfocado más en un diseño sencillo del código generando sólo la documentación indispensable.
2. Comunicación: potenciada por el desarrollo en pares y la presencia del cliente, además de la simplicidad en cuanto al código.
3. Retroalimentación: propiciada por el protagonismo del cliente que participa activamente y por el trabajo en ciclos cortos.
4. Coraje: enfrentando decisiones en ocasiones complejas que pudieran afectar el tiempo de desarrollo y la calidad del producto.
5. Respeto: basado en estimar en toda su magnitud el trabajo de los demás.

Ventajas

- Es apropiado para entornos volátiles.
- Se está preparado para el cambio, significando una reducción en su coste.
- La planificación es más transparente para los clientes ya que conocen las fechas de entrega de las funcionalidades vitales para su negocio.
- Permite definir en cada iteración cuáles son los objetivos de la siguiente.
- Permite tener una retroalimentación por parte de los usuarios.
- La presión está a lo largo de todo el proyecto y no en una entrega final. (JEFFRIES)

Después de todo el estudio previamente realizado se puede llegar a la conclusión de que históricamente, las metodologías tradicionales han intentado abordar la mayor



cantidad de situaciones de contexto principalmente a proyectos de gran magnitud, ya que exigen un gran esfuerzo para ser adaptadas, sobre todo en proyectos pequeños y con requisitos muy cambiantes, como la arquitectura a desarrollar. Sin embargo las metodologías ágiles ofrecen una solución casi a la medida para una gran cantidad de proyectos con características similares a las de la arquitectura que se desea implementar, sobre todo XP que, aunque para algunos resulta muy arriesgada su utilización por una serie de inconvenientes y restricciones, es la que mejor se ajusta a las necesidades y condiciones de un equipo de trabajo compuesto por sólo dos personas y que cuenta con un corto período de tiempo para el desarrollo de una solución final. Todo esto es posible gracias a la sencillez que presenta XP, tanto en su aprendizaje como en su aplicación, reduciendo los costos de implantación en un equipo de desarrollo.

1.5 Conclusiones del capítulo.

En el presente capítulo se realizó una investigación sobre el estado del arte, se ha podido apreciar todo lo referente a la base teórica que fundamenta esta investigación, dando a conocer al lector los principales conceptos que se manejan, las diferentes tecnologías para el desarrollo de aplicaciones, los sistemas automatizados que existen en el mundo vinculados con el campo donde se proyecta el objeto de estudio, así como la fundamentación del lenguaje de programación a utilizar, el Sistema Gestor de Base de Datos y la Metodología que se eligió para darle solución al problema planteado.



CAPÍTULO II - PLANEACIÓN Y DISEÑO

2.1 Introducción

En este capítulo, se introduce la fase de planeación y diseño, donde se detallan las necesidades del cliente, se describen las funcionalidades que serán objeto de automatización mediante el empleo de las **historias de usuario** (HU), se realiza una estimación del esfuerzo necesario para las mismas y se establece un plan de iteraciones necesarias sobre el sistema para su terminación. Además, se presentan las tarjetas Clases, Responsabilidades y Colaboradores, que permitirán trabajar con una metodología basada en objetos.

2.2 Funcionalidades generales

En el ISMMM el centro de red de computadoras del mismo presta a sus usuarios una serie de servicios informáticos los cuales para su asignación deben pasar por un estricto proceso de aprobación. Cada solicitud debe estar aprobada por cada uno de los facultativos pertinentes; para poder realizar la asignación de los servicios solicitados.

De la descripción antes expuesta, se identifican como requisitos generales del sistema, *el análisis del estado del Proceso de Solicitud y Aprobación de Servicios Informáticos y el control de su ejecución.*

Análisis del estado del Proceso de Solicitud y Aprobación de Servicios Informáticos y el control de su ejecución.

Consiste en darle seguimiento a todas las solicitudes hechas desde cada uno de los departamentos del centro, a través de la generación de reportes, así como eliminar las mismas, además debe permitir la aprobación de dichas solicitudes por los facultativos y mostrar estas solicitudes aprobadas para su asignación, dar baja de utilización de servicios, enviar avisos de solicitudes pendientes, etc.



2.2.1 Personal relacionado con el sistema

Tabla 2.1 Personas relacionadas con el sistema

Personas relacionadas con el sistema	Justificación
Especialista	Esta es la persona que tiene conocimiento del Proceso de Solicitud y Aprobación de Servicios Informáticos, y está encargada de la gestión de información de la misma.
Administrador	Es la persona encargada de asesorar y dar seguimiento del estado del proceso de desarrollo.
Desarrollador	Es la persona responsable de llevar a cabo la implementación del sistema.

2.2.2 Lista de reserva

Después de conocer el personal relacionado e identificar los requisitos generales, se procede a realizar el análisis de las funcionalidades que debe cumplir la aplicación para darle respuesta a los mismos. Para ello se enumerarán mediante una lista de reserva, las funcionalidades que el sistema debe ser capaz de cumplir ([Anexo 2](#)).

2.2.3 Historias de usuario

Las HU, son la técnica utilizada en XP para detallar los requisitos del software. Son el resultado directo del intercambio entre los usuarios y desarrolladores a través de reuniones donde las conocidas *tormenta de ideas (brain storm)* arrojan no solo los requerimientos, sino también las posibles soluciones; representan una forma rápida



de administrar las necesidades de los usuarios sin tener que elaborar gran cantidad de documentos formales y sin requerir de mucho tiempo para gestionarlos, debido a que un requerimiento de software es descrito de forma concreta y sencilla utilizando el lenguaje común del usuario. Las HU permiten responder ágilmente a los requerimientos cambiantes y aunque se redactan desde las perspectivas de los clientes, también los desarrolladores pueden brindar ayuda en la identificación de las mismas. Para definir las se emplea la siguiente plantilla. (Aguilera, 2010)

Tabla 2.2 Planilla de historia de usuario

Historia de usuario	
Número: No. Historia de usuario	Usuario: Usuario entrevistado para obtener la función requerida a automatizar.
Nombre: nombre de la historia de usuario que sirve para identificarla mejor entre los desarrolladores y el cliente.	
Prioridad en el negocio: Importancia: Alta / Media / Baja	Riesgo en desarrollo: Dificultad: Alta / Media / Baja
Puntos estimados: Estimación: de 1 a 3 puntos	Iteración asignada: Iteración a la que corresponde
Programador responsable: Nombre de encargado de programación.	
Descripción: Se especifican las operaciones por parte del usuario y las respuestas del sistema.	
Observaciones: Algunas observaciones de interés, como glosario, información sobre usuario etc.	

A continuación se muestran las historias de usuario descritas por el cliente:

Tabla 2.3 HU No.1: Configurar Sistema

Historia de usuario	
Número: 1	Usuario: Administrador del Sistema
Nombre: Configurar Sistema	
Prioridad en el negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 3	Iteración asignada: 1
Programador responsable: Yasmany Chávez cruz.	
Descripción: El usuario debe insertar las facultades y departamentos del centro, los servicios que se van a prestar y los usuarios que interactuarán con el sistema. Una vez añadida esta información, el sistema te permite mostrar, modificar y eliminar la misma.	
Observaciones: Confirmado con el cliente.	

Para ver las historias de usuario ver [\[ANEXO 3: HISTORIAS DE USUARIO\]](#).

2.3 Planificación de entregas

En esta fase se establece la prioridad de cada HU, y a continuación, se realiza una estimación del esfuerzo necesario de cada una de ellas por parte de los programadores. Se toman acuerdos sobre el contenido de la primera entrega y se determina un cronograma en conjunto con el cliente. Una entrega debe obtenerse en no más de dos a tres meses.(Aguilera, 2010)

Las estimaciones asociadas a la implementación de las historias se establecen empleando como medida el punto de estimación. Un punto de estimación equivale a

una semana ideal de programación, donde los miembros de los equipos de desarrollo trabajan el tiempo planeado sin ningún tipo de interrupción, este punto de estimación que se utiliza para representar la semana ideal, es de 5 días. Las historias generalmente tienen un valor de 1 a 3 puntos. Además, se mantiene un registro de la velocidad de desarrollo, establecida por puntos de iteración, basado fundamentalmente en la suma de puntos de estimación correspondientes a las HU, que fueron terminadas en la última iteración.(Aguilera, 2010)

2.3.1 Estimación de esfuerzo por historias de usuario

Para el buen desarrollo del sistema propuesto, se realizó una estimación para cada una de las HU identificadas, y se obtienen los resultados que se muestran a continuación:

Tabla 2.4 Estimación de esfuerzo por historia de usuario

Historias de usuario	Puntos de estimación
Configurar Sistema	3 semanas
Autenticar usuarios	4 días
Gestionar datos de solicitud de servicios de un usuario individual	2 semanas
Gestionar datos de solicitud de servicios de los Estudiantes	2 semanas
Gestionar autorizo por el decano	1 semana
Gestionar autorizo por el V.R.I.P	1 semana
Reportar solicitudes aprobadas	2 semanas
Gestión para asignar servicios	4 días
Gestionar bajas de utilización de servicios	1 semana
Saber estado de la solicitud	3 días

2.3.2 Planificación de iteraciones

A partir de las HU antes expuestas y la estimación del esfuerzo propuesto para la realización de las mismas, se procede a realizar la planificación de la etapa de implementación del sistema, apoyándose en el tiempo e intentando concentrar las funcionalidades relacionadas en una misma iteración. En este plan se establece cuántas iteraciones serán necesarias realizar sobre el sistema para su terminación. El plan de iteraciones puede contener indicaciones sobre cuáles HU se incluirán en un release (versión, distribución), lo cual debería ser consistente con el contenido de una o dos iteraciones.(Aguilera, 2010)

En relación con lo antes tratado se decide realizar el sistema en 5 iteraciones, las cuales se explican de forma detalla a continuación:

Primera iteración:

Esta iteración tiene como objetivo darle cumplimiento a las HU.1 y 2 que se consideraron de mayor importancia para el desarrollo de la aplicación en la HU.2 los usuarios deben autenticarse en el sistema y el sistema permite su entrada o envía un mensaje de error en el caso de que los usuarios introduzcan los datos mal; en la HU.1 se configurara el sistema insertando los datos necesarios para el buen funcionamiento del mismo. Al concluir dicha iteración se contará con todas las funcionalidades descritas en las HU 1 y 2, donde en la misma el usuario debe insertar las facultades y departamentos del centro, también los servicios informáticos que se prestan en dicho centro y además los usuarios que interactuarán con el sistema y gestionarán el proceso de solicitudes. Una vez añadida esta información, se podrá mostrar, modificar y eliminar la misma.

Se tendrá la primera versión de prueba, que contará con dos modelos de desarrollo que incorporan todas las funcionalidades antes vistas, éstos modelos se le presentarán al cliente con el objetivo de obtener una retroalimentación del mismo para posteriores iteraciones del producto.



Segunda iteración:

Esta iteración, no menos importante que la primera, tiene como finalidad desarrollar las HU 3 y 4. En esta el usuario debe insertar solicitudes de servicios pero existen dos tipos de solicitudes: 1-para un usuario individual y 2-para un grupo de usuarios con un pedido en común (estudiantes) y estas solicitudes se manejan cada una de forma diferente por eso se utiliza una HU para cada tipo de solicitud, el sistema te permite guardar en base de datos dichas solicitudes, mostrar las mismas, etc. La versión que se obtenga de esta iteración en unión con la entregada en la iteración anterior se le facilitará al cliente para comprobar si cumple con las necesidades antes acordadas con él.

Tercera iteración:

Esta iteración tiene como propósito llevar a cabo el desarrollo de la HU 6 y 7. Donde las solicitudes hechas deben pasar por dos escalones de aprobación, primero por el Decano y las que el Decano aprueba pasan por el V.R.I.P que es el que da la aprobación final por eso el manejo de esta información se dividen en dos HU distintas; donde aquí en caso del Decano dado un listado de usuarios con servicios solicitados por los administradores de los departamentos, autoriza o deniega parte o todos los servicios solicitados, y el sistema te permite guardar dicho autorizo de solicitud, además de enviar automáticamente un aviso de Solicitudes Pendientes al V.R.I.P. ; y en caso del V.R.I.P dado un listado de usuarios con servicios solicitados por los decanos, autoriza o deniega parte o todos los servicios solicitados, y el sistema te permite guardar dicho autorizo de solicitud, además de enviar automáticamente un aviso de Solicitudes Pendientes al Administrador de Red. Y la misma se facilita al cliente para comprobar si cumple con las necesidades antes acordadas con él.

Cuarta iteración:

Esta iteración tiene como propósito llevar a cabo el desarrollo de la HU 8; donde el usuario dado un listado de solicitudes con servicios asignados, manda al sistema a

reportar planillas de solicitud, y luego si el usuario desea puede mandar a imprimir dicha planilla a través del navegador; además el sistema permitirá también al usuario visualizar el listado de bajas de utilización de servicios. Esta historia de usuario será integrada con el resultado de las iteraciones anteriores y se le facilitará al cliente para comprobar si cumple con las necesidades antes acordadas con él.

Quinta iteración:

Esta iteración tiene como propósito llevar a cabo el desarrollo de la HU 9, 5 y 10. Donde en el caso de HU 9 el usuario le hace una petición al sistema de listar usuarios y sus servicios aprobados, y el sistema le ofrece dicho listado, luego del usuario haber asignado los servicios aprobados acepta dicho listado como listado de usuarios con sus servicios asignados y el sistema guarda dicho listado. En caso de la HU 5 el usuario dado un listado de usuarios con servicios otorgados de su departamento, selecciona los usuarios a dar baja de utilización de servicios, y el sistema debe permitir guardar un listado de usuarios bajas de utilización de servicios. Y la HU 10 está diseñada para que los usuarios que realizan solicitudes individuales puedan ver en qué nivel de aprobación esta su solicitud donde el usuario inserta su carné de identidad y manda al sistema a buscar, y el sistema realiza un búsqueda y si encuentra dicha solicitud le muestra al usuario el nombre, el estado de la solicitud y los servicios.

Estas historias de usuario serán integradas con el resultado de las iteraciones anteriores, y como fruto de esta integración se obtendrá la versión 1.0 del producto final. A partir de este momento el software será puesto a un proceso de prueba para evaluar el desempeño del mismo.

2.3.3 Plan de duración de las iteraciones

Como parte del ciclo de vida de un proyecto guiado por la metodología de desarrollo de software XP, se crea el plan de duración de cada una de las iteraciones que se llevarán a cabo durante el desarrollo del mismo. Este plan tiene como finalidad



mostrar la duración de cada iteración, así como el orden en que serán implementadas las HU en cada una de las mismas.

Tabla 2.5 Plan de duración de las iteraciones

Iteración	Historias de usuario	Duración total
Iteración 1	Configurar Sistema	3 semanas
	Autenticar usuarios	
Iteración 2	Gestionar datos de solicitud de servicios de un usuario individual	4 semanas
	Gestionar datos de solicitud de servicios de los Estudiantes	
Iteración 3	Gestionar bajas de utilización de servicios.	1 semana
Iteración 4	Gestionar autorizo por el decano	4 semanas
	Gestionar autorizo por el V.R.I.P	
	Reportar solicitudes aprobadas	
Iteración 5	Gestión para asignar servicios	1 semana
	Saber estado de la solicitud	

2.4 Clases, responsabilidades y colaboradores

En este epígrafe tiene lugar la realización de las tarjetas de clases, responsabilidades y colaboración, conocidas tradicionalmente como *tarjetas CRC*



([Anexo 4](#)), las cuáles se realizan con el objetivo de facilitar la comunicación y documentar los resultados. Además, las mismas permiten la total participación y contribución del equipo de desarrollo en la tarea de diseño. Una tarjeta CRC representa un objeto, por tanto es una clase, cuyo nombre se ubica en forma de título en la parte superior de la tarjeta, los atributos y las responsabilidades más significativas se colocan a la izquierda y las clases implicadas con cada responsabilidad a la derecha, en la misma línea de su requerimiento correspondiente. Para mejor comprensión de las mismas, se determina agruparlas por HU.(Aguilera, 2010)

Tabla 2.6 Tarjeta CRC No.1

Nombre de la clase: Administrar Sistema	
Tipo de la clase: Lógica del negocio	
Responsabilidades:	Colaboradores:
Insertar Facultades.	Consulta _ modelo, FunsGlobs.php, database.php
Mostrar Facultades.	
Modificar Facultades.	
Eliminar Facultades.	
Insertar Departamentos.	
Mostrar Departamentos.	
Modificar Departamentos.	
Eliminar Departamentos.	
Insertar Servicios.	
Mostrar Servicios.	
Modificar Servicios.	
Eliminar Servicios.	
Insertar Usuarios.	
Mostrar Usuarios.	
Modificar Usuarios.	
Eliminar Usuarios.	



2.5 Conclusiones

En este capítulo se abordó la fase de planeación y diseño donde se delinearon las HU con la participación del cliente, se llevó a cabo la planificación de iteraciones de cada HU a partir de la estimación del esfuerzo necesario de las mismas. Presentando además, las principales clases mediante el empleo de las tarjetas CRC, culminando así esta fase y se determina que el equipo de trabajo está listo para pasar a la siguiente etapa de desarrollo.



CAPITULO III – DESARROLLO Y PRUEBAS

3.1 Introducción

En este capítulo se inicia la fase de desarrollo y pruebas conforme a la metodología XP. Se presenta el modelo de datos empleado para la aplicación concluyente, y se realiza el desarrollo de las iteraciones a partir del desglose de las historias de usuario en tareas. Así como también aparecen las interfaces gráficas de usuario diseñadas para la aplicación final. Se describen además las pruebas realizadas y se indican las respuestas de la aplicación en el empleo de las diferentes funcionalidades, así como los posibles mensajes de error, información o confirmación que emite el sistema cuando se utiliza una de estas funcionalidades.

3.2 Modelo de datos

En esta parte se muestra el modelo de datos empleado para la aplicación.

Primero se tiene una tabla niveles en la cual a través de un procedimiento almacenado el programador inserta los roles o niveles de acceso que van a tener los usuarios que interactuarán con el sistema; que estos van a ser: administrador, Admin_Departamento, Decano, V.R.I.P y Admin_redes.

Luego se configura el sistema previéndolo de los datos necesarios para su correcto funcionamiento; donde primero se gestiona facultades insertando dichos datos en la tabla facultades, una facultad tiene varios departamentos y se utiliza una tabla llamada departamentos para almacenar los datos de los mismos; también se utiliza una tabla usuarios para guardar los datos de los usuarios del sistema y esta tiene relación(1-1) con las tablas facultades y departamentos respectivamente; además se

necesita gestionar también datos de los servicios que se prestan en el centro y para eso se cuenta con una tabla servicios donde se registran el nombre de los mismos.

Por otra parte, los Admin_Departamentos realizan solicitudes a sus respectivos Decanos y para gestionar las mismas, estas son guardadas en las tablas solicitudes_Indiv o solicitud_Estud respectivamente en dependencia de la solicitud que realice, resaltar aquí que una solicitud de estudiantes tiene un grupo de estudiantes que piden los mismos servicios y para esto se crea una nueva tabla llamada “Estudiantes” para guardar los datos individuales de cada estudiante que pertenece a un grupo.

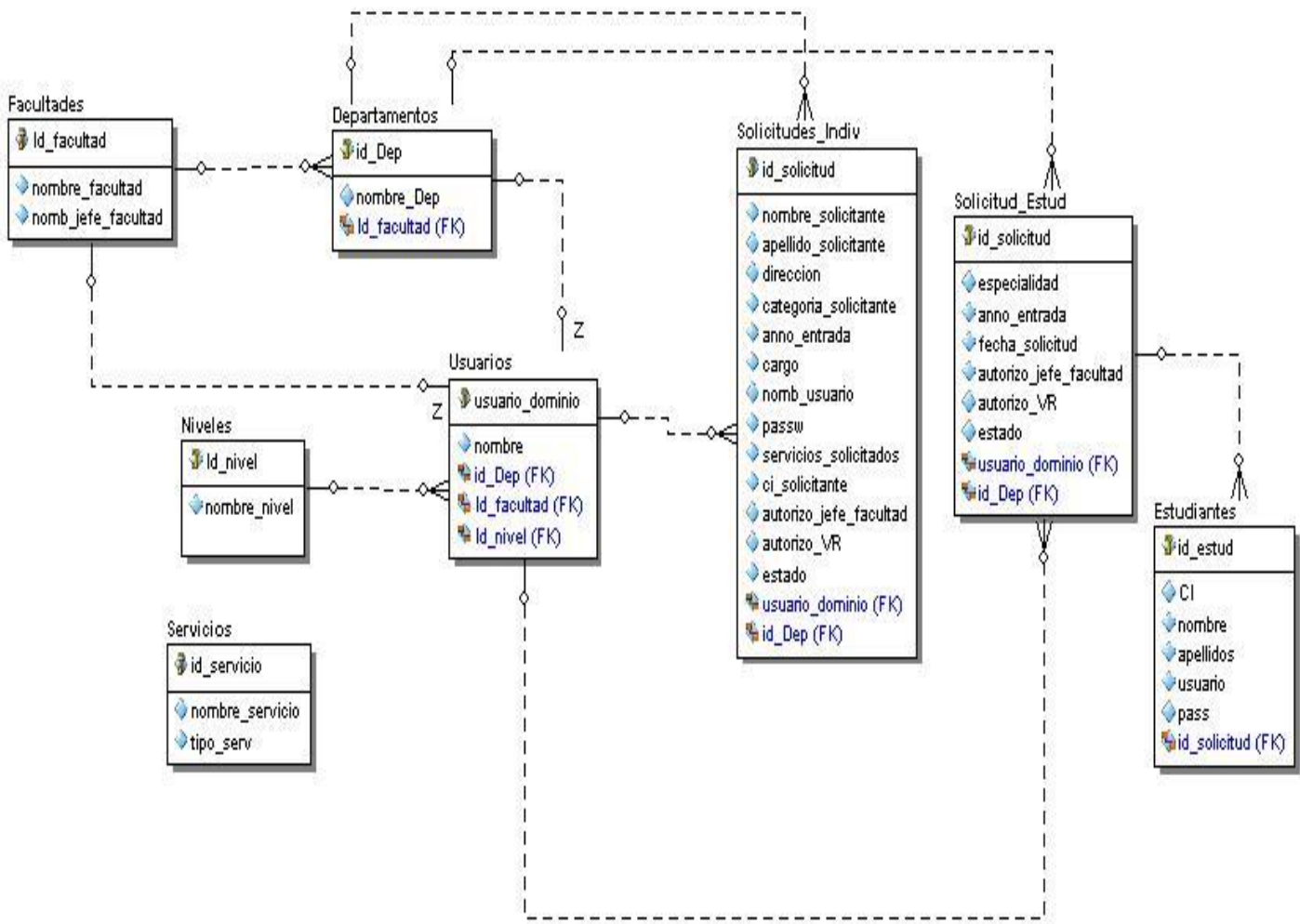




Fig. 3.1 Modelo de datos

3.3 Desarrollo de las iteraciones

Durante la fase planificación y diseño fueron detalladas las historias de usuario correspondientes a cada una de las iteraciones a desarrollar, teniendo en cuenta las prioridades y restricciones de tiempo, previstas por el cliente.

3.3.1 Tareas por historias de usuario

Dentro del contenido de este plan, las HU se descomponen en tareas de programación o ingeniería, y a su vez, estas son asignadas al equipo de desarrollo para su implementación. Las tareas no tienen que ser entendidas necesariamente por el cliente, pues las mismas, sólo son utilizadas por los miembros del equipo de desarrollo, por lo que pueden ser escritas en lenguaje técnico. Las mismas se representan mediante las tarjetas de tareas.



Tabla 3.1 Distribución de tareas por historia de usuario

Historia de usuario	Tareas
Configurar Sistema	-Insertar Facultades -Modificar Facultades -Insertar Departamentos -Modificar Departamentos -Insertar Servicios -Modificar Servicios -Insertar Usuarios -Modificar Usuarios
	-Mostrar Facultades. -Eliminar Facultades. -Mostrar Dptos. -Eliminar Dptos. -Mostrar Servicios. -Eliminar Servicios. -Mostrar Usuarios. -Eliminar Usuarios.
Autenticar usuarios	-Verificar nombre de usuario y contraseña.
Gestionar datos de solicitud de servicios de un usuario individual	-Insertar solicitud de usuario individual. -Eliminar solicitud. -Aviso de Solicitudes Pendientes al Decano
Gestionar datos de solicitud de servicios de los Estudiantes	-Insertar solicitud de estudiantes. -Eliminar solicitud. -Aviso de Solicitudes Pendientes al Decano.
Gestionar autorizo por el Decano	-Listar usuarios servicios solicitados. -Guardar autorizo solicitud de servicios. -Aviso Solicitudes Pendientes al V.R.I.P.
Gestionar autorizo por el V.R.I.P	-Listar usuarios servicios solicitados. -Guardar autorizo solicitud de servicios. -Aviso Solicitudes Pendientes al Admin. de Red.
Reportar solicitudes aprobadas	-Listar solicitudes con servicios asignados. -Mostrar planilla de solicitud. -Listar usuarios bajas de utilización de servicios
Gestión para asignar servicios	-Lista de usuarios con servicios aprobados. -Guardar como lista de usuarios con servicios asignados.
Gestionar bajas de utilización de servicios.	-Listar usuarios servicios otorgados. -Guardar listado de usuarios bajas.
Saber estado de la solicitud	-Mostrar estado de la solicitud.

Tabla 3.2 Historias de usuario abordadas en la primera iteración

Historias de usuario	Tiempo de estimación (semanas)	
	Estimación inicial	Real
Configurar Sistema	3	3
Autenticar usuarios	0.3	0.3

Tabla 3.3 Historias de usuario abordadas en la segunda iteración

Historias de usuario	Tiempo de estimación (semanas)	
	Estimación inicial	Real
Gestionar datos de solicitud de servicios de un usuario individual	2	1.5
Gestionar datos de solicitud de servicios de los Estudiantes	2	1.5

Tabla 3.4 Historias de usuario abordadas en la tercera iteración

Historias de usuario	Tiempo de estimación (semanas)	
	Estimación inicial	Real
Gestionar autorizo por el Decano	1	1
Gestionar autorizo por el V.R.I.P	1	1

Tabla 3.5 Historias de usuario abordadas en la cuarta iteración

Historias de usuario	Tiempo de estimación (semanas)	
	Estimación inicial	Real
Reportar solicitudes aprobadas	2	2

Tabla 3.6 Historias de usuario abordadas en la quinta iteración

Historias de usuario	Tiempo de estimación (semanas)	
	Estimación inicial	Real
Gestión para asignar servicios	0.5	0.5
Gestionar bajas de utilización de servicios	1	1
Saber estado de la solicitud	0.4	0.4

Para ver las tarjetas de ingeniería ver [\[ANEXO 6: TARJETAS DE TAREAS\]](#).

3.4 Pruebas

En la Programación Extrema es esencial el desarrollo de las pruebas, permitiendo probar continuamente el código. Cada vez que se desea implementar las funcionalidades que tendrá el software, XP propone una redacción sencilla de prueba, para ser pasada por el código posteriormente. El proceso constante de las pruebas permite la obtención un producto con mayor calidad ofreciendo a los programadores una mayor certeza en el trabajo que desempeñan. En la metodología XP hay dos tipos de pruebas; las unitarias o desarrollo dirigido por pruebas (TDD Test Driven Development), desarrolladas por los programadores verificando su código de forma automática, y las pruebas de aceptación, las cuáles son evaluadas luego de culminar una iteración verificando así que se cumplió la funcionalidad requerida por el cliente. Con estas normas se obtiene un código simple y funcional de manera bastante rápida y eficiente. Por esto es importante pasar las pruebas al 100%. (Aguilera, 2010)



3.4.1 Desarrollo dirigido por pruebas

El desarrollo dirigido por pruebas, se enfoca en la implementación orientada a pruebas. El código debe ser probado paso a paso para lograr un resultado, aunque no con lógica para el negocio, pero si funcional. Algunas personas confunden este término con las llamadas “pruebas de caja blanca” las cuáles se les practican a los métodos u operaciones para medir la funcionalidad del mismo, desde el punto de vista de validez del cliente. Sin embargo, el TDD se aplica antes de comenzar a implementar cada paso de la tarea en desarrollo, asumiendo que la prueba es insatisfactoria desde un inicio. Sólo una vez que se haya cumplido de la forma más sencilla posible la lógica del código a probar se asume como cumplida. Luego se realiza un proceso conocido como “refactorización” de código perteneciente a una de las doce prácticas planteadas por la metodología XP, el cual consiste en mantener el código en buen estado, modificándolo activamente para que conserve claridad y sencillez. Es esencia el TDD, se enfoca en la lógica del negocio y las pruebas de caja blanca en la lógica del negocio. (Aguilera, 2010)

3.4.2 Pruebas de Aceptación

Las pruebas de aceptación en XP, se pueden asociar con las pruebas de caja negra que se aplican en otras metodologías de desarrollo, sólo que se crean a partir de las historias de usuario y no por un listado de requerimientos. Durante las iteraciones, las HU se traducen a pruebas de aceptación. En ellas se especifican desde la perspectiva del cliente, los escenarios para probar que una historia de usuario ha sido implementada correctamente. La misma puede tener todas las pruebas de aceptación que necesite para asegurar su correcto funcionamiento. El objetivo que persiguen estas pruebas, es garantizar que las funcionalidades solicitadas por el cliente han sido realizadas. Una HU no se considera completa hasta que no ha transitado por sus pruebas de aceptación. Luego de ver los paradigmas anteriores empleados para la realización de las pruebas y reunirse con el cliente para su



análisis, el mismo decidió que se lleve a cabo el proceso mediante las pruebas de aceptación.(Aguilera, 2010)

La planilla utilizada para plasmar el contenido de las pruebas de aceptación se muestra a continuación.

Tabla 3.7 Planilla de prueba de aceptación

Prueba de aceptación
HU: Nombre de la historia de usuario que va a comprobar su funcionamiento.
Nombre: Nombre del caso de prueba.
Descripción: Descripción del propósito de la prueba.
Condiciones de ejecución: Precondiciones para que la prueba se realice.
Entrada/Pasos ejecución: Pasos para probar la funcionalidad.
Resultado: Resultado que se desea de la prueba.
Evaluación de la prueba: Aceptada o denegada.

Para más información de las pruebas de aceptación ver [\[ANEXO 7: PRUEBAS DE ACEPTACION\]](#).



3.5 Conclusiones

En este capítulo se llevó a cabo la fase de desarrollo y prueba donde se presenta el modelo de datos de la aplicación a construir, logrando una visión detallada de sus atributos y las relaciones entre sus clases. Se realizó el desarrollo de las iteraciones a partir de la distribución de tareas por HU, y se les practicó las pruebas de aceptación a las funcionalidades de mayor importancia.



CAPITULO IV - ESTUDIO DE FACTIBILIDAD

4.1 Introducción

La probabilidad de restricciones, insuficiencia de recursos y presiones en las fechas de entregas implícitas en el desarrollo en un proyecto, es real, por lo que es ineludible y sensato evaluar la viabilidad del mismo. De identificarse en la fase de definición, posibles riesgos en la concepción del proyecto, es válido minimizar de forma razonable recursos humanos, materiales y financieros. En general los productos informáticos no están exentos de estas dificultades, por lo que es de vital importancia estimar la relación costo-beneficio, así como el esfuerzo, capital humano y el tiempo de desarrollo que se demanda en la ejecución de los mismos.(Aguilera, 2010)

4.2 Métodos para la estimación del esfuerzo

Análisis de puntos de estimación y COCOMO II - El SEI (del inglés, Software Engineering Institute), propone desde hace algunos años un método para la estimación del esfuerzo llamado COCOMO II (Construction Cost Model). El mismo está basado en ecuaciones matemáticas que permiten calcular el esfuerzo a partir de ciertas métricas de tamaño estimado, como el análisis de puntos de función y las líneas de código fuente (en inglés SLOC, Source Lines of Code).(Aguilera, 2010)

Análisis de puntos de casos de uso - Existe una posibilidad de predecir el tamaño de un sistema a partir de las características de sus requisitos, expresados en los casos de uso. La estimación mediante el análisis de puntos de casos de uso, es un método propuesto originalmente por Gustav Karner de Objectory AB, y posteriormente refinado por muchos otros autores. Se trata de un método de estimación del tiempo de desarrollo de un proyecto mediante la asignación de



“pesos” a un cierto número de factores que lo afectan, para finalmente, contabilizar el tiempo total estimado para el proyecto a partir de esos factores.(Aguilera, 2010)

4.3 Estudio de factibilidad

Para la estimación del proyecto en la investigación se emplea el método del análisis de puntos de casos de uso (CU), ya que el sistema cuenta con 10 HU que es el equivalente a los CU en la metodología XP, siendo esta una vía sencilla que cuenta con un software denominado Estimac (Cálculo de la estimación de proyectos informáticos basado en puntos de casos de uso) para proporcionar su uso.

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

El primer paso para la estimación consiste en el cálculo de *puntos de casos de uso sin ajustar*, este valor se obtiene a partir de la siguiente ecuación:

$$UUCP = UAW + UUCW$$

Donde:

UUCP (Unadjusted Use Cases Points): puntos de CU sin ajustar.

UAW (Unadjusted Actors Weight): factor de peso de los actores sin ajustar.

UUCW (Unadjusted Use Cases Weight): factor de peso de los CU sin ajustar.



Factor de peso de los actores sin ajustar (UAW)

Este valor se calcula mediante un análisis de la cantidad de actores presentes en el sistema y la complejidad de cada uno de ellos. La complejidad de los actores se establece teniendo en cuenta en primer lugar si se trata de una persona o un sistema, y en segundo lugar la forma en que el actor interactúa con el sistema.

Tabla 4.1 Factor de peso de los actores sin ajustar.

Tipo	Peso	Valor	Peso*Valor
Simple	1	0	0
Medio	2	0	0
Complejo	3	5	15
UAW: 15			

Factor de peso de los casos de uso sin ajustar (UUCW)

Este valor se calcula mediante un análisis de la cantidad de CU presentes en el sistema y la complejidad de cada uno de ellos. La complejidad de los CU se establece teniendo en cuenta la cantidad de transacciones efectuadas en el mismo, donde una transacción se entiende como una secuencia de actividades atómicas, es decir, se efectúa la secuencia de actividades completa, o no se efectúa ninguna de las actividades de la secuencia.



Tabla 4.2 Factor de peso de los CU sin ajustar.

Tipo	Peso	Valor	Peso*Valor
Simple	5	7	35
Medio	10	2	20
Complejo	15	1	15
UUCW: 70			

Finalmente los puntos de CU sin ajustar resultaron:

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

$$\text{UUCP} = 15 + 70$$

$$\text{UUCP} = 85$$

Una vez que se tienen los puntos de CU sin ajustar, se debe ajustar este valor calculando los puntos de CU ajustados mediante la ecuación:

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

Donde:

UCP (Use Cases Points): puntos de CU ajustados.

UUCP (Unadjusted Use Cases Points): puntos de CU sin ajustar.

TCF (Technical Complexity Factor): factor de complejidad técnica.

EF (Environment Factor): factor de ambiente.



Factor de complejidad técnica

Este coeficiente se calcula mediante la cuantificación de un conjunto de factores que determinan la complejidad técnica. Cada uno de estos factores se cuantifica con un valor de 0 a 5, donde 0 significa un aporte irrelevante y 5 un aporte muy importante.

El factor de complejidad técnica se calcula mediante la ecuación:

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

$$TCF = 0.6 + 0.01 \times 18$$

$$TCF = 0.78$$

Tabla 4.3 Factor de complejidad técnica.

Factor	Peso	Valor	Peso*Valor
T1	2	1	2
T2	1	2	2
T3	1	2	2
T4	1	1	1
T5	1	2	2
T6	0.5	1	0.5
T7	0.5	1	0.5
T8	2	1	2
T9	1	2	2
T10	1	2	2
T11	1	1	1
T12	1	1	1
T13	1	0	0
Total			18
TCF: 0.78			



Factor de ambiente

Las habilidades y el entrenamiento del grupo involucrado en el desarrollo tienen un gran impacto en las estimaciones de tiempo. Estos factores son los que se contemplan en el cálculo del factor de ambiente. El cálculo del mismo es similar al del factor de complejidad técnica, es decir, se trata de un conjunto de factores que cuantifican con valores de 0 a 5.

El factor ambiente se calcula:

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

$$EF = 1.4 - 0.03 \times 20.5$$

$$EF = 0.785$$

Tabla 4.4 Factor de ambiente.

Factor	Peso	Valor	Peso*Valor
E1	1.5	3	4.5
E2	0.5	3	1.5
E3	1	3	3
E4	0.5	3	1.5
E5	1	3	3
E6	2	4	8
E7	-1	0	0
E8	-1	1	-1
Total			20.5
EF: 0.785			



Finalmente los puntos de CU ajustados resultan:

$$\text{UCP} = 85 * 0.78 * 0.785$$

$$\text{UCP} = 52.0455$$

Luego el esfuerzo en horas-hombre viene dado por:

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

$$\text{E} = 52.0455 * 20 \text{ h/hombre}$$

$$\text{E} = 1040.91$$

Donde:

E (Effort): esfuerzo estimado en horas-hombre.

UCP (Use Cases Points): puntos de CU ajustados.

CF (Conversion Factor): factor de conversión.

Este método proporciona una estimación del esfuerzo en horas-hombre contemplando sólo el desarrollo de la funcionalidad especificada en los CU. Para una estimación más exacta de la duración completa del proyecto, hay que agregar a la estimación del esfuerzo obtenida por los puntos de CU, las estimaciones de esfuerzo de las demás actividades relacionadas con el desarrollo de software.



Tabla 4.5 Estimación del esfuerzo total del proyecto.

Actividad	% Esfuerzo	Valor esfuerzo
Análisis	10	260.2275
Diseño	20	520.455
Implementación	40	1040.91
Prueba	15	390.34125
Sobrecarga	15	390.34125
Total	100%	2602.275

El proyecto requiere de 2602.275 horas-hombres para su desarrollo, por tanto si trabaja 1 persona, 280 horas como promedio en un mes que equivale a: 70 horas como promedio en un semana=>10 horas como promedio en un día, el proyecto se puede terminar en aproximadamente 278.7 días (9.29meses)

Costo por Mes (CM) y Costo General (CG):

CM = Salario Total

CM = Salario * cantidad de obreros

CM = 100.00 * 1

CM = 100.00 Pesos por mes.

CG = CM * TDEV

CG = 100.00 * 9.29 meses

CG = 929.00 Pesos

El desarrollo del sistema tendrá un costo estimado de \$ 929.00.



4.4 Análisis de sostenibilidad

Cuando un sistema informático se instaura, éste repercute de forma positiva y/o negativa en los usuarios finales, por lo que es necesario efectuar el análisis de su impacto en las dimensiones administrativas, socio-humanistas, ambientales y tecnológicas, para contribuir al desarrollo sostenible del producto informático. Se ha afirmado con frecuencia que las nuevas tecnologías informáticas tendrán un impacto económico tan fuerte como el de la revolución industrial, de hecho, gracias a los progresos de las tecnologías informáticas y de las comunicaciones, el modo en que los directivos gestionan la información en sus organizaciones donde trabajan, ha cambiado completamente, pasando de los informes impresos al uso de programas informáticos que les ofrezcan la información que precisan.(Aguilera, 2010)

El desarrollo de este sistema no supone grandes gastos de recursos, ni de tiempo; la base de datos que contiene puede ser alojada en los sistemas de gestión existentes en el centro, ya que los mismos tienen buenas prestaciones y acceso rápido. La tecnología empleada para el desarrollo del sistema es totalmente libre, no requiere incurrir en gastos adicionales, en el pago de licencias de uso. Es portable por lo que un cambio de plataforma para la implantación del mismo es viable y factible, y no hay que infringir en muchos cambios.

Incluso cuando existe un gasto de electricidad asociado al consumo de los ordenadores y el servidor del sistema debe mantenerse encendido 24 horas, el mismo no aumentará el consumo actual al estar considerado en los niveles de gastos.



4.5 Beneficios tangibles e intangibles

El sistema para la gestión de solicitudes de servicios informáticos en el ISMMM, no es un producto con fines comerciales, su principal objetivo es resolver los problemas que existen durante el desarrollo de este proceso en el centro. El beneficio fundamental del sistema es contar con una aplicación Web flexible, dinámica y de interfaz agradable que le permita gestionar, aprobar y conocer de una forma más precisa y en el menor tiempo posible las solicitudes de servicios informáticos del centro; a las personas relacionadas con el proceso.

Con el aprovechamiento de este sistema que se desarrolla, se consiguen beneficios directos al medio ambiente, ya que disminuye considerablemente el volumen de papel y tinta antes usado por el personal que gestiona el proceso, eliminando así el uso excesivo de estos recursos.

Los beneficios inmediatos por lo general son los siguientes:

- Disminución del tiempo y esfuerzo que se invierte en esta tarea realizada hasta el momento de forma manual.
- Disminución de la acumulación de materiales impresos relacionados con los procesos de solicitudes de servicios, etc.
- Ahorro de tiempo en la búsqueda de información de las solicitudes.
- Los usuarios pueden revisar si sus solicitudes fueron aprobadas.
- Seguridad al guardar la información.
- El cliente ya cuenta con un sistema capaz de guardar grandes volúmenes de información de forma segura, eliminando los inconvenientes de que se pierdan o deterioren y que se acumulen de forma excesiva.



- El centro dispone de toda la información relacionada con las solicitudes de servicios informáticos de forma automatizada, lo que implica mayor seguridad y confiabilidad.

Unido al sistema se debe entregar un manual de usuario donde se podrán consultar en detalle el funcionamiento del sistema. Los tiempos de respuestas del sistema son mínimos y los principales retrasos están dados por el tráfico de la red, por lo que los usuarios podrán ejecutar sus operaciones de forma rápida. La implementación de la aplicación no tiene incidencias desfavorables sobre el medio ambiente.

Existe una estrecha comunicación entre los desarrolladores de la aplicación y los usuarios finales, logrando que el módulo que se presenta, satisfaga las necesidades y expectativas de los clientes finales.

Una vez efectuado el análisis del producto informático se puede arribar a las siguientes **conclusiones del capítulo**:

- El producto desplegado ofrece respuesta a la necesidad que lo origina.
- Los recursos para su desarrollo son mínimos ya que se realiza con los disponibles y las tecnologías sobre las que funciona están basadas en software libre.
- Se emplea la filosofía de trabajo con que se realizan los sistemas en el centro, lo que facilitará su mantenimiento y futuras actualizaciones.
- La solución que se propone es flexible a cambios.



CONCLUSIONES GENERALES

El sistema se desarrolla siguiendo la metodología XP, y se utilizaron representaciones para la modelación de todas las fases del proyecto. El sistema resultante está provisto de un ambiente cómodo, fácil de entender, que cumple los estándares de diseño y utiliza técnicas modernas de programación orientada a objetos. Después de realizado el trabajo se arriba a las siguientes conclusiones:

1. Se elaboró el marco teórico metodológico que fundamenta la investigación, permitiendo el análisis del proceso de gestión y aprobación de solicitudes de servicios informáticos en el centro para lograr una mejor comprensión del negocio.
2. Se efectuó un estudio de las diferentes tecnologías y herramientas para la confección del sistema, haciendo una elección, de acuerdo a las especificaciones del cliente.
3. Se efectuó un levantamiento de los requerimientos, los que propiciaron un mejor entendimiento de la problemática en cuestión, facilitando la planificación, el diseño e implementación del sistema automatizado, el cual solucionó las dificultades antes expuestas.
4. Se desarrolló e implementó toda una investigación donde se obtuvo una aplicación Web que permite la gestión y aprobación de solicitudes de servicios informáticos en el centro.



RECOMENDACIONES

De manera general los objetivos trazados al inicio de esta investigación han sido logrados, al mismo tiempo, en el transcurso del proceso de desarrollo, ha quedado evidenciado, que la propuesta es sólo la primera fase de un proyecto que puede ser mucho más ambicioso. Por tanto se hacen las siguientes recomendaciones.

1. Implantar el sistema en el centro para su utilización.
2. Incentivar la creación de nuevas herramientas para continuar el desarrollo del Sistema Automatizado para la Gestión y Aprobación de Solicitudes de Servicios Informáticos en el ISMMM.
3. Implementar nuevas funcionalidades que enriquezcan el proceso de gestión y aprobación de solicitudes de servicios informáticos en el centro.



REFERENCIAS BIBLIOGRÁFICAS

Aguilera, J. M. B. (2010). SISTEMA DE GESTION INTEGRAL DE LA EMPRESA EMPLEADORA DEL NIQUEL - MODULO GESTION DE CONTRATOS DE COMPRAS., ISMM.

BLANCO CRIADO, A. (2008). "XAMPP, 2008. [en línea]." Disponible en:
<http://www.adictosaltrabajo.com/tutoriales>.

"Embarcadero ER/Studio. [en línea]." [Consultado: 2011-02-25] Disponible en:
http://bureaudeprensa.com/es/view.php?bn=bureaudeprensa_software&key=1153755975.

. "ER/Studio. [en línea]." Disponible en:
<http://www.monografias.com/trabajos14/modelodebase/modelodebase.shtml>.

FERRER, J. "Metodologías Ágiles. [en línea]." Disponible en:
<http://libresoft.es/downloads/ferrer-20030312.pdf>.

JEFFRIES, R. "What is Extreme Programming? [en línea]." [Consultado: 2011-04-15]. Disponible en: <http://www.xprogramming.com/xpmag/whatisxp.htm>.

Lozano, V. Code Igniter
En español, Conocimiento Virtual Academia Ltda.
Myer, T. Profesional CodeIgniter, Wiley Publishing, Inc.

"MySQL. [en línea]." [Consultado: 2011-03-10] Disponible en:
<http://es.wikipedia.org/wiki/MySQL>.

. "Paquete Preconfigurado XAMPP [en Línea]." Disponible en:
<http://www.genbeta.com/2005/09/27-xampp-servidor-web-correo-base-de-datos-y-mas-extras-en-pocos-minutos>.

"PHP Designer 2008 v6.0.2.0 Professional. [en línea]." [Consultado 2011-03-10] Disponible en:
<http://www.intercambiosvirtuales.org/software/php-designer-2008-v6020-professional>.

"Sitio Web de la Universidad de Oviedo." [Consultado: 2011-04-5] Disponible en:
<http://www.dgr.gub.uv/gestionusuarios.asp.htm>

. "XAMPP. [en línea]." Disponible en: <http://e.wikipedia.org/wiki/XAMPP>.



BIBLIOGRAFÍAS

Aplicaciones Web. [en Línea]. [Consultado: 2011-06-02]. Disponible en: <http://www.iec.csic.es/criptonomicon/javascript/javascript.html>

Aplicaciones Web. [en Línea]. [Consultado: 2011-06-02]. Disponible en: <http://www.todoexpertos.com/quick/>

Aplicaciones Web. [en línea]. [Consultado: 2011-01-15]. Disponible en: <http://www.forosdelweb.com>

Aplicaciones web. [en línea]. [Consultado: 2011-01-15]. Disponible en: <http://www.todoexpertos.com>

Aplicaciones web. [en Línea]. [Consultado: 2011-06-02]. Disponible en: <http://www.bolsasistemas.com>

Aplicaciones web. [en Línea]. [Consultado: 2011-06-02]. Disponible en: <http://foros.cristalab.com>

Aplicaciones web. [en Línea]. [Consultado: 2011-06-02]. Disponible en: <http://www.webtaller.com>

Aplicaciones web. [en Línea]. [Consultado: 2011-06-02]. Disponible en: <http://www.trabajoline.com.ar>

MONMANY, J. Aplicaciones Web. [en línea]. [Consultado: 2011-01-15]. Disponible en: <http://www.webvillage.info>

GLOSARIO DE TERMINOS

HTML: Hyper Text Markup Language, o simplemente HTML, es un lenguaje de programación muy sencillo que se utiliza para crear los textos y las páginas web. Si se hace la traducción de su nombre del inglés al castellano, sería “Lenguaje de Marca de Hipertextos”, ya que es justamente un lenguaje que se basa en las marcas para crear los hipertextos.

Multiplataforma: Es un término usado para referirse a los programas, sistemas operativos, lenguajes de programación, u otra clase de software, que puedan funcionar en diversas plataformas.

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.

V.R.I.P: Vicerrector de Investigación y Postgrado.

Admin_redes: Usuario del sistema que tiene el rol de Administrador de Redes que es el encargado de asignar los servicios.

Admin_Departamento: Usuario del sistema que tiene el rol de Administrador del Departamento que es el encargado de insertar las solicitudes.

solicitudes_Indiv: Se refiere a la tabla Solicitudes Individuales de la base de dato.

solicitudes_estud: Se refiere a la tabla Solicitudes de Estudiantes de la base de dato.

HU: Historia de Usuario.

ANEXO 1: Planillas de solicitud de Servicios**Planilla de solicitud de servicios informáticos de un usuario individual.**

Nombres:			
Apellidos:			
Carné de Identidad:			
Dirección Particular:			
Categoría:	(Si es estudiante) Año:		
Área:			
Dpto:			
Nombre del Jefe del área:			
Cargo:			

Nombre de Usuario (buzón, Internet, Intranet):			
--	--	--	--

Contraseñas de los distintos servicios, puede ser la misma para todo:			
Email:		Internet:	
Intranet:			

Tipo de servicios a usar: (Debe de llevar la firma de los decanos o jefes de área en cada servicio autorizado)

_____ ☐ Correo Nacional

_____ ☐ Correo Internacional

_____ ☐ Servicios de acceso total a la INTRANET

_____ ☐ Servicios de acceso total a INTERNET a través de las PC de mi Dpto. o Facultad.

_____ ☐ Todos los servicios.

Manifiesto que los datos arriba expuestos se ajustan a la realidad y registraré mi comportamiento como usuario de la red ISMMM. Doy fe de que he leído los capítulos del Reglamento de la RED ISMMM al dorso de este Formulario, así como su Código de Ética Interno y que por lo tanto acataré todo lo expresado por la letra de dichos documentos así como también las disposiciones que estipule su Administración.

Me comprometo además a no divulgar o usar a través de la RED ISMMM hacia el exterior ninguna información que por su carácter sea confidencial, clasificada o de importancia estratégica en cualquier sentido.

Moa, 15 de enero de 2011

Firma del Solicitante	Firma y Cuño del Jefe de área	Dr. Allan Pierra Conde Funcionario que autoriza (VRIP)
-----------------------	-------------------------------	---

Figura 1.1 Planilla de solicitud de servicios (Usuario Individual).

[illegible]

Moa, 15 de enero de 2011

Firma y Cuiño del Jefe de área	Dr. Allan Pierra Conde Funcionario que autoriza (VRIP)
--------------------------------	---

Figura 1.2 Planilla de solicitud de servicios (Estudiantes).

ANEXO 2: Continuación de la lista de reserva del producto

- Insertar Facultades.
- Mostrar Facultades.
- Eliminar Facultades.
- Modificar Facultades.
- Insertar Departamentos.
- Mostrar Departamentos.
- Eliminar Departamentos.
- Modificar Departamentos.
- Insertar Servicios.
- Mostrar Servicios.
- Eliminar Servicios.
- Modificar Servicios.
- Insertar Usuarios.
- Mostrar Usuarios.
- Eliminar Usuarios.
- Modificar Usuarios.
- Insertar solicitud de servicios de un usuario individual.
- Eliminar solicitud de servicios de un usuario individual.
- Enviar aviso de **Solicitudes Pendientes** al Decano.
- Insertar solicitud de servicios de un grupo de usuarios que tienen un pedido en común (Estudiantes).
- Eliminar solicitud de servicios de estudiantes.
- Listar usuarios con servicios asignados.
- Dar baja de utilización de servicios.
- Guardar listado de usuarios dados de bajas.
- Mostrar listado de usuarios dados de bajas.
- Listar solicitudes hechas por los Administradores de Departamentos.
- Guardar autorizo por el Decano de solicitudes de servicios.
- Enviar aviso de **Solicitudes Pendientes** al V.R.I.P.



- Listar solicitudes hechas por los decanos.
- Guardar autorizo por el V.R.I.P de solicitudes de servicios.
- Enviar aviso de **Solicitudes Pendientes** Administrador de Red.
- Listar solicitudes aprobadas.
- Guardar como solicitudes asignadas
- Listar solicitudes asignadas.
- Reportar plantilla de solicitud asignada.
- Mostrar estado de la solicitud.

ANEXO 3: HISTORIAS DE USUARIO

HU No.2: Autenticar Usuarios

Historia de usuario	
Número: 2	Usuario: Usuarios del Sistema(después lo pongo)
Nombre: Autenticar Usuarios	
Prioridad en el negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 3	Iteración asignada: 1
Programador responsable: Yasmany Chávez cruz.	
Descripción: Para poder entrar al sistema los usuarios que interactúan con el deben entrar usuario y contraseña y el sistema verifica que los datos estén correctos en caso contrario el sistema envía un mensaje de error.	
Observaciones: Confirmado con el cliente.	

HU No.3: Gestionar datos de solicitud de servicios (usuario individual)

Historia de usuario	
Número: 3	Usuario: Administrador del Departamento
Nombre: Gestión de solicitudes(usuario individual)	
Prioridad en el negocio: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 3	Iteración asignada: 2
Programador responsable: Yasmany Chávez Cruz.	
Descripción: El usuario debe insertar cada una de las solicitudes hechas por el personal de su departamento, el sistema te permite mostrar las mismas y eliminarlas además de enviar un aviso automáticamente(a través de correo) al decano, de solicitudes pendientes.	
Observaciones: Confirmado con el cliente.	

**HU No.4:** Gestionar datos de solicitud de servicios de los estudiantes

Historia de usuario	
Número: 4	Usuario: Administrador del Departamento
Nombre: Gestionar solicitud de estudiantes.	
Prioridad en el negocio: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 3	Iteración asignada: 2
Programador responsable: Yasmany Chávez Cruz.	
Descripción: Para facilitarle el trabajo al usuario el puede ir a la Intranet2 del centro y utilizar el servicio de "Recuperador del SIGENU" donde esta toda la información de los estudiantes del centro y copiar los datos que le hagan falta para la solicitud y introducirlos al formulario del sistema y el sistema te permitirá mostrar y eliminar las solicitudes insertadas, además de enviar un aviso automáticamente(a través de correo) al decano, de solicitudes pendientes.	
Observaciones: Confirmado con el cliente.	

HU No.5: Gestionar bajas de utilización de servicios

Historia de usuario	
Número: 5	Usuario: Administrador del Departamento
Nombre: Gestionar bajas de utilización de servicios	
Prioridad en el negocio: Media	Riesgo en desarrollo: Alto
Puntos estimados: 2	Iteración asignada: 5
Programador responsable: Yasmany Chávez Cruz.	
Descripción: El usuario dado un listado de usuarios con servicios asignados de su departamento, selecciona los usuarios a dar baja de utilización de servicios, y el sistema debe permitir guardar este listado de usuarios bajas.	
Observaciones: Confirmado con el cliente.	

HU No.6: Gestionar autorizo por el decano

Historia de usuario	
Número: 6	Usuario: Decano
Nombre: Gestionar autorizo por el decano	
Prioridad en el negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 2	Iteración asignada: 3
Programador responsable: Yasmany Chávez Cruz.	
Descripción: El usuario dado un listado de usuarios con servicios solicitados, por los Administradores de los Departamentos, autoriza o deniega parte o todos los servicios solicitados, y el sistema te permite guardar dicho autorizo de solicitud, además de enviar automáticamente un aviso de Solicitudes Pendientes al V.R.I.P.	
Observaciones: Confirmado con el cliente.	

HU No.7: Gestionar autorizo por el V.R.I.P

Historia de usuario	
Número: 7	Usuario: Vicerrector de Investigación y Postgrado
Nombre: Gestionar autorizo por el decano	
Prioridad en el negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 2	Iteración asignada: 3
Programador responsable: Yasmany Chávez Cruz.	
Descripción: El usuario dado un listado de usuarios con servicios solicitados por los decanos, autoriza o deniega parte o todos los servicios solicitados, y el sistema te permite guardar dicho autorizo de solicitud, además de enviar automáticamente un aviso de Solicitudes Pendientes al Administrador de Red.	
Observaciones: Confirmado con el cliente.	

**HU No.8:** Reportar solicitudes aprobadas.

Historia de usuario	
Número: 8	Usuario: Administrador de Redes
Nombre: Reportar solicitudes aprobadas	
Prioridad en el negocio: Media	Riesgo en desarrollo: Alto
Puntos estimados: 2	Iteración asignada: 4
Programador responsable: Yasmany Chávez Cruz.	
Descripción: El usuario dado un listado de solicitudes aprobadas manda a mostrar la planilla de solicitud de servicio deseada, además el sistema debe permitir listar los usuarios dados de baja por los administradores de departamentos.	
Observaciones: Confirmado con el cliente.	

HU No.9: Gestión para asignar servicios

Historia de usuario	
Número: 9	Usuario: Administrador de Redes
Nombre: Gestión para asignar servicios	
Prioridad en el negocio: Baja	Riesgo en desarrollo: Alto
Puntos estimados: 2	Iteración asignada: 5
Programador responsable: Yasmany Chávez Cruz.	
Descripción: El usuario le hace una petición al sistema de listar usuarios y sus servicios aprobados, y el sistema le ofrece dicho listado, luego del usuario haber asignado los servicios aprobados acepta dicho listado como listado de usuarios con sus servicios asignados y el sistema guarda dicho listado.	
Observaciones: Confirmado con el cliente.	



HU No.10: Saber estado de la Solicitud.

Historia de usuario	
Número: 10	Usuario: Usuarios que realizan Solicitudes Individuales
Nombre: Saber estado de la Solicitud	
Prioridad en el negocio: Media	Riesgo en desarrollo: Alto
Puntos estimados: 2	Iteración asignada: 5
Programador responsable: Yasmany Chávez Cruz.	
Descripción: El usuario debe insertar en el sistema su carné de identidad y presionar el botón buscar, si el usuario realizó la solicitud el sistema debe mostrar el nombre del usuario, en que nivel de aprobación está la solicitud y los servicios en cada caso.	
Observaciones: Confirmado con el cliente.	

ANEXO 4: TARJETAS CRC

Tarjeta CRC No.2: Autenticar usuarios

Nombre de la clase: Autenticar usuarios	
Tipo de la clase: Lógica del negocio	
Responsabilidades:	Colaboradores:
Verificar nombre de usuario	Consulta _ modelo, Validaciones, FunsGlobs.php, database.php
Verificar contraseña(Para entrar al Sistema)	

Tarjeta CRC No.3: Gestionar datos de solicitud de servicios de un usuario individual

Nombre de la clase: Gestionar datos de solicitud de servicios de un usuario individual	
Tipo de la clase: Lógica del negocio	
Responsabilidades:	Colaboradores:
Insertar solicitud de usuario individual.	Consulta _ modelo, Validaciones, FunsGlobs.php, Datos.php, database.php
Eliminar solicitud	
Aviso de Solicitudes Pendientes al Decano	

Tarjeta CRC No.4: Gestionar datos de solicitud de servicios de los Estudiantes

Nombre de la clase: Gestionar datos de solicitud de servicios de los Estudiantes	
Tipo de la clase: Lógica del negocio	
Responsabilidades:	Colaboradores:
Insertar solicitud de estudiantes	Consulta _ modelo, Validaciones, FunsGlobs.php, Datos.php, database.php
Eliminar solicitud	
Aviso de Solicitudes Pendientes al Decano	

Tarjeta CRC No.5: Gestionar bajas de utilización de servicios

Nombre de la clase: Gestionar bajas de utilización de servicios	
Tipo de la clase: Lógica del negocio	
Responsabilidades:	Colaboradores:
Listar usuarios servicios otorgados.	Consulta _ modelo, database.php
Guardar listado usuarios bajas.	

Tarjeta CRC No.6: Gestionar autorizo por el decano

Nombre de la clase: Gestionar autorizo por el decano	
Tipo de la clase: Lógica del negocio	
Responsabilidades:	Colaboradores:
Listar usuarios servicios solicitados.	Consulta _ modelo, Datos.php, database.php
Guardar autorizo solicitud de servicios.	
Aviso Solicitudes Pendientes al V.R.I.P.	

Tarjeta CRC No.7: Gestionar autorizo por el V.R.I.P

Nombre de la clase: Gestionar autorizo por el V.R.I.P	
Tipo de la clase: Lógica del negocio	
Responsabilidades:	Colaboradores:
Listar usuarios servicios solicitados.	Consulta _ modelo, Datos.php, database.php
Guardar autorizo solicitud de servicios.	
Aviso Solicitudes Pendientes al Admin. de Red.	

Tarjeta CRC No.8: Reportar solicitudes aprobadas

Nombre de la clase: Reportar solicitudes aprobadas	
Tipo de la clase: Lógica del negocio	
Responsabilidades:	Colaboradores:
Listar Solicitudes servicios asignados.	Consulta _ modelo, Datos.php, database.php
Mostrar planilla de solicitud.	
Listar usuarios bajas	

Tarjeta CRC No.9: Gestión para asignar servicios

Nombre de la clase: Gestión para asignar servicios	
Tipo de la clase: Lógica del negocio	
Responsabilidades:	Colaboradores:
Lista de usuarios con servicios aprobados.	Consulta _ modelo, Datos.php, database.php
Guardar como usuarios con servicios asignados.	

Tarjeta CRC No.10: Saber estado de la Solicitud.

Nombre de la clase: Mostrar listado de usuarios con sus servicios asignados	
Tipo de la clase: Lógica del negocio	
Responsabilidades:	Colaboradores:
Mostrar estado de la solicitud	Consulta _ modelo, Datos.php, database.php



ANEXO 6: TARJETAS DE TAREAS

Para la primera Interacción

Tarjeta de tarea No.1: Gestionar datos de Facultades

Tarea ingeniería	
Número tarea: 1	Número historia: 1
Nombre tarea: Gestionar datos de Facultades.	
Tipo de tarea : Desarrollo	Puntos estimados: 1
Fecha inicio: 1/02/2011	Fecha fin: 6/02/2011
Programador responsable: Yasmany Chávez Cruz.	
Descripción: Esta tarea facilita introducir los datos de las facultades, mostrar, modificar y eliminar los mismos.	

Tarjeta de tarea No. 2: Gestionar datos de Departamentos

Tarea de ingeniería	
Número tarea: 2	Número historia: 1
Nombre tarea: Gestionar datos de Departamentos.	
Tipo de tarea : Desarrollo	Puntos estimados: 1
Fecha inicio:6/02/2011	Fecha fin: 11/02/2011
Programador responsable: Yasmany Chávez Cruz.	
Descripción: Esta tarea facilita introducir los datos de los Departamentos, mostrar, modificar y eliminar los mismos.	

**Tarjeta de tarea No. 3:** Gestionar datos de Usuarios

Tarea de ingeniería	
Número tarea: 3	Número historia: 1
Nombre tarea: Gestionar datos de Usuarios.	
Tipo de tarea : Desarrollo	Puntos estimados: 3
Fecha inicio: 11/02/2011	Fecha fin: 16/02/2011
Programador responsable: Yasmany Chávez Cruz.	
Descripción: Esta tarea facilita introducir los datos de los Usuarios, mostrar, modificar y eliminar los mismos.	

Tarjeta de tarea No. 4: Gestionar datos de Servicios

Tarea de ingeniería	
Número tarea: 4	Número historia: 1
Nombre tarea: Gestionar datos de Servicios.	
Tipo de tarea : Desarrollo	Puntos estimados: 2
Fecha inicio: 16/02/2011	Fecha fin: 21/02/2011
Programador responsable: Yasmany Chávez Cruz.	
Descripción: Esta tarea facilita introducir los datos de los Usuarios, mostrar, modificar y eliminar los mismos.	



Tarjeta de tarea No. 5: Verificar nombre de usuario y contraseña.

Tarea de ingeniería	
Número tarea: 5	Número historia: 2
Nombre tarea: Verificar nombre de usuario y contraseña.	
Tipo de tarea : Desarrollo	Puntos estimados: 1
Fecha inicio: 21/02/2011	Fecha fin: 25/02/2011
Programador responsable: Yasmany Chávez Cruz.	
Descripción: Esta tarea permite la entrada de los usuarios o envía un mensaje de error en el caso de que los usuarios introduzcan los datos incorrectos.	

Para la Segunda Interacción:

Tarjeta de tarea No. 6: Insertar solicitud de usuario individual.

Tarea de ingeniería	
Número tarea: 6	Número historia: 3
Nombre tarea: Insertar solicitud de usuario individual.	
Tipo de tarea : Desarrollo	Puntos estimados: 1
Fecha inicio: 25/02/2011	Fecha fin: 1/03/2011
Programador responsable: Yasmany Chávez Cruz.	
Descripción: Esta tarea te permite insertar cada una de las solicitudes hechas por el personal de su departamento.	

**Tarjeta de tarea No. 7:** Eliminar solicitud de usuario individual.

Tarea de ingeniería	
Número tarea: 7	Número historia: 3
Nombre tarea: Eliminar solicitud de usuario individual.	
Tipo de tarea : Desarrollo	Puntos estimados: 1
Fecha inicio: 1/03/2011	Fecha fin: 5/03/2011
Programador responsable: Yasmany Chávez Cruz.	
Descripción: Esta tarea facilita eliminar cada una de las solicitudes hechas por el personal de su departamento	

Tarjeta de tarea No. 8: Aviso de Solicitudes Pendientes (Usuario Individual)

Tarea de ingeniería	
Número tarea: 8	Número historia: 3
Nombre tarea: Aviso de Solicitudes Pendientes al Decano.	
Tipo de tarea : Desarrollo	Puntos estimados: 1
Fecha inicio: 5/03/2011	Fecha fin: 9/03/2011
Programador responsable: Yasmany Chávez Cruz.	
Descripción: Esta tarea facilita enviar un aviso automáticamente(a través de correo) al decano, de solicitudes pendientes.	

**Tarjeta de tarea No. 9:** Insertar solicitud de estudiantes

Tarea de ingeniería	
Número tarea: 9	Número historia: 4
Nombre tarea: Insertar solicitud de estudiantes.	
Tipo de tarea : Desarrollo	Puntos estimados: 1
Fecha inicio: 9/03/2011	Fecha fin: 13/03/2011
Programador responsable: Yasmany Chávez Cruz.	
Descripción: Esta tarea te permite insertar cada una de las solicitudes de estudiantes echas por los grupos de su departamento.	

Tarjeta de tarea No. 10: Eliminar solicitud de estudiantes.

Tarea de ingeniería	
Número tarea: 10	Número historia: 4
Nombre tarea: Eliminar solicitud de estudiantes.	
Tipo de tarea : Desarrollo	Puntos estimados: 1
Fecha inicio: 13/03/2011	Fecha fin: 17/03/2011
Programador responsable: Yasmany Chávez Cruz.	
Descripción: Esta tarea te permite eliminar cada una de las solicitudes de estudiantes echas por los grupos de su departamento.	

**Tarjeta de tarea No. 11:** Aviso de Solicitudes Pendientes (Estudiantes)

Tarea de ingeniería	
Número tarea: 11	Número historia: 4
Nombre tarea: Aviso de Solicitudes Pendientes al Decano.	
Tipo de tarea : Desarrollo	Puntos estimados: 1
Fecha inicio: 17/03/2011	Fecha fin: 21/03/2011
Programador responsable: Yasmany Chávez Cruz.	
Descripción: Esta tarea facilita enviar un aviso automáticamente(a través de correo) al decano, de solicitudes pendientes.	

Para la Tercera Interacción:**Tarjeta de tarea No. 12:** Listar usuarios con servicios solicitados

Tarea de ingeniería	
Número tarea: 12	Número historia: 6
Nombre tarea: Listar usuarios servicios solicitados.	
Tipo de tarea : Desarrollo	Puntos estimados: 2
Fecha inicio: 21/03/2011	Fecha fin: 22/03/2011
Programador responsable: Yasmany Chávez Cruz.	
Descripción: Permite al Decano, listar usuarios con servicios solicitados por los Administradores de los Departamentos y autorizar o denegar parte o todos los servicios solicitados.	



Tarjeta de tarea No. 13: Guardar autorizo de solicitud de servicios de los Decanos

Tarea de ingeniería	
Número tarea: 13	Número historia: 6
Nombre tarea: Guardar autorizo solicitud de servicios.	
Tipo de tarea : Desarrollo	Puntos estimados: 2
Fecha inicio: 22/03/2011	Fecha fin: 25/03/2011
Programador responsable: Yasmany Chávez Cruz.	
Descripción: Permite guardar autorizo de solicitud servicios de los decanos.	

Tarjeta de tarea No. 14: Aviso Solicitudes Pendientes al V.R.I.P

Tarea de ingeniería	
Número tarea: 14	Número historia: 6
Nombre tarea: Guardar autorizo solicitud de servicios.	
Tipo de tarea : Desarrollo	Puntos estimados: 2
Fecha inicio: 25/03/2011	Fecha fin: 26/03/2011
Programador responsable: Yasmany Chávez Cruz.	
Descripción: Permite enviar automáticamente un aviso de Solicitudes Pendientes al V.R.I.P.	

Tarjeta de tarea No. 16: Listar usuarios con servicios autorizados por los Decanos

Tarea de ingeniería	
Número tarea: 16	Número historia: 7
Nombre tarea: Listar usuarios con servicios autorizados por los Decanos.	
Tipo de tarea : Desarrollo	Puntos estimados: 2
Fecha inicio: 28/03/2011	Fecha fin: 29/03/2011
Programador responsable: Yasmany Chávez Cruz.	
Descripción: Permite dado un listado de usuarios con servicios autorizados por los decanos, autoriza o deniega parte o todos los servicios solicitados.	

Tarjeta de tarea No. 17: Guardar autorizo solicitud del V.R.I.P

Tarea de ingeniería	
Número tarea: 17	Número historia: 7
Nombre tarea: Guardar autorizo solicitud del V.R.I.P.	
Tipo de tarea : Desarrollo	Puntos estimados: 2
Fecha inicio: 29/03/2011	Fecha fin: 31/03/2011
Programador responsable: Yasmany Chávez Cruz.	
Descripción: Permite guardar dicho autorizo de solicitud.	

**Tarjeta de tarea No. 18:** Aviso Solicitudes Pendientes

Tarea de ingeniería	
Número tarea: 18	Número historia: 7
Nombre tarea: Aviso Solicitudes Pendientes.	
Tipo de tarea : Desarrollo	Puntos estimados: 2
Fecha inicio: 31/03/2011	Fecha fin: 1/04/2011
Programador responsable: Yasmany Chávez Cruz.	
Descripción: Permite enviar automáticamente un aviso de Solicitudes Pendientes al Administrador de Red.	

Para la Cuarta Iteración:**Tarjeta de tarea No. 20:** Listar Solicitudes con servicios asignados.

Tarea de ingeniería	
Número tarea: 20	Número historia: 8
Nombre tarea: Listar Solicitudes con servicios asignados.	
Tipo de tarea : Desarrollo	Puntos estimados: 2
Fecha inicio: 3/04/2011	Fecha fin: 9/04/2011
Programador responsable: Yasmany Chávez Cruz.	
Descripción: Esta tarea permite mostrar un listado de solicitudes con servicios asignados.	

**Tarjeta de tarea No. 21:** Mostrar Planilla de Solicitud

Tarea de ingeniería	
Número tarea: 21	Número historia: 8
Nombre tarea: Mostrar Planilla de Solicitud.	
Tipo de tarea : Desarrollo	Puntos estimados: 2
Fecha inicio: 9/04/2011	Fecha fin: 15/04/2011
Programador responsable: Yasmany Chávez Cruz.	
Descripción: Esta tarea permite reportar una solicitud en una planilla en el navegador y luego si el usuario desea puede mandar a imprimir dicha planilla a través del navegador.	

Tarjeta de tarea No. 22: Listar bajas de utilización de servicios

Tarea de ingeniería	
Número tarea: 22	Número historia: 8
Nombre tarea: Listar bajas de utilización de servicios.	
Tipo de tarea : Desarrollo	Puntos estimados: 1
Fecha inicio: 15/04/2011	Fecha fin: 17/04/2011
Programador responsable: Yasmany Chávez Cruz.	
Descripción: Esta tarea permite reportar una solicitud en una planilla en el navegador y luego si el usuario desea puede mandar a imprimir dicha planilla a través del navegador.	

**Para la Quinta Iteración:**

Tarjeta de tarea No. 23: Lista de usuarios con servicios aprobados.

Tarea de ingeniería	
Número tarea: 23	Número historia: 9
Nombre tarea: Lista de usuarios con servicios aprobados.	
Tipo de tarea : Desarrollo	Puntos estimados: 2
Fecha inicio: 17/04/2011	Fecha fin: 22/04/2011
Programador responsable: Yasmany Chávez Cruz.	
Descripción: Esta tarea permite listar usuarios y sus servicios aprobados.	

Tarjeta de tarea No. 24: Guardar listado, como usuarios con servicios asignados

Tarea de ingeniería	
Número tarea: 24	Número historia: 9
Nombre tarea: Guardar listado, como usuarios con servicios asignados.	
Tipo de tarea : Desarrollo	Puntos estimados: 2
Fecha inicio: 22/04/2011	Fecha fin: 27/04/2011
Programador responsable: Yasmany Chávez Cruz.	
Descripción: Esta tarea permite aceptar dicho listado como listado de usuarios con sus servicios asignados y guardar dicho listado.	



Tarjeta de tarea No. 25: Listar usuarios con servicios asignados.

Tarea de ingeniería	
Número tarea: 25	Número historia: 5
Nombre tarea: Listar usuarios con servicios asignados.	
Tipo de tarea : Desarrollo	Puntos estimados: 2
Fecha inicio: 27/04/2011	Fecha fin: 1/05/2011
Programador responsable: Yasmany Chávez Cruz.	
Descripción: Esta tarea permite listar los usuarios con servicios asignados del departamento del que lo mande a listar.	

Tarjeta de tarea No. 26: Guardar listado de usuarios dados bajas.

Tarea de ingeniería	
Número tarea: 26	Número historia: 5
Nombre tarea: Listar usuarios con servicios asignados.	
Tipo de tarea : Desarrollo	Puntos estimados: 1
Fecha inicio: 1/05/2011	Fecha fin: 2/05/2011
Programador responsable: Yasmany Chávez Cruz.	
Descripción: Esta tarea permite guardar el listado de usuarios bajas que el Administrador del Departamento seleccionó.	



Tarjeta de tarea No. 26: Saber estado de la solicitud

Tarea de ingeniería	
Número tarea: 26	Número historia: 10
Nombre tarea: Saber estado de la solicitud.	
Tipo de tarea : Desarrollo	Puntos estimados: 2
Fecha inicio: 2/05/2011	Fecha fin: 6/05/2011
Programador responsable: Yasmany Chávez Cruz.	
Descripción: Esta tarea permite saber en que estado está la solicitud individual.	

ANEXO 7: PRUEBAS DE ACEPTACIÓN

Pruebas de aceptación para la HU: Configurar Sistema

Esta se va dividir en 4 pruebas de aceptación para comprobar mejor el funcionamiento de la misma ya que dentro de esta se recogen 4 funcionalidades más que son:

- Gestionar Facultades.
- Gestionar Departamentos.
- Gestionar Servicios.
- Gestionar Usuarios.

Pruebas de aceptación para la HU: comprobar la entrada de datos de las facultades.

Prueba de aceptación
HU: Configurar Sistema.
Nombre: Prueba para comprobar la entrada de datos de las facultades.
Descripción: Validación de entrada de datos de contrato.
Condiciones de ejecución: El usuario debe introducir todos los datos pedidos en el formulario facultades.
Entrada/Pasos ejecución: El usuario intenta introducir los datos de las facultades.
Resultado: Se emite un mensaje de error en caso de que: • Se introduzcan campos vacíos en el formulario facultades. • Se intente insertar una facultad que ya existe. • Se inserte un jefe de facultad que ya es jefe de una facultad. • En caso contrario se emite un mensaje de éxito en la inserción.
Evaluación de la prueba: Aceptada.

Pruebas de aceptación para la HU: comprobar la entrada de datos de los departamentos.

Prueba de aceptación
HU: Configurar Sistema.
Nombre: Prueba para comprobar la entrada de datos de los departamentos.
Descripción: Validación de entrada de datos de los departamentos.
Condiciones de ejecución: El usuario debe introducir los datos pedidos de los departamentos.
Entrada/Pasos ejecución: El usuario intenta introducir los datos del departamento.
Resultado: Se emite un mensaje de error en caso de: • Se introduzcan campos vacíos en el formulario departamentos. • Se intente insertar un departamento que ya existente. • En caso contrario se emite un mensaje de éxito en la inserción.
Evaluación de la prueba: Aceptada.

Pruebas de aceptación para la HU: comprobar la entrada de datos de los servicios.



Prueba de aceptación
HU: Configurar Sistema.
Nombre: Prueba para comprobar la entrada de datos de los servicios.
Descripción: Validación de entrada de datos de los servicios.
Condiciones de ejecución: El usuario debe introducir los datos pedidos de los servicios.
Entrada/Pasos ejecución: El usuario intenta introducir los datos de los servicios.
Resultado: Se emite un mensaje de error en caso de: • Se introduzcan campos vacíos en el formulario servicios. • Se intente insertar un servicio que ya existente. • En caso contrario se emite un mensaje de éxito en la inserción.
Evaluación de la prueba: Aceptada.

Pruebas de aceptación para la HU: comprobar la entrada de datos de los usuarios

Prueba de aceptación
HU: Configurar Sistema.
Nombre: Prueba para comprobar la entrada de datos de los usuarios.
Descripción: Validación de entrada de datos de los usuarios.
Condiciones de ejecución: El usuario debe introducir los datos pedidos de los usuarios.
Entrada/Pasos ejecución: El usuario intenta introducir los datos de los usuarios.
Resultado: Se emite un mensaje de error en caso de: • Los campos usuario y Nombre Completo sean vacíos. • Se intente insertar un usuario que ya existente. • Se intente insertar más de un usuario con el nivel V.R.I.P. • Se intente insertar varios administradores de departamento de un mismo departamento en el sistema. • Se intente insertar un decano de una misma facultad. • En caso contrario se emite un mensaje de éxito en la inserción.
Evaluación de la prueba: Aceptada.

Pruebas de aceptación para la HU: Autenticar usuarios

Prueba de aceptación
HU: Autenticar usuarios.
Nombre: Prueba para comprobar la entrada al sistema de los usuarios que interactúan con el.
Descripción: Validación de entrada de usuarios al sistema.
Condiciones de ejecución: El usuario debe insertar usuario y contraseña.
Entrada / Pasos ejecución: El usuario intenta entrar al sistema.
Resultado: Se emite un mensaje de error en caso de: • Los campos sean vacíos. • Se verifique en la base de datos el usuario y no sea usuario del sistema. • Se verifique contra el correo y no coincida el usuario y la contraseña insertada por el usuario, con la de usuario y contraseña del correo.
Evaluación de la prueba: Aceptada.

Pruebas de aceptación para la HU: Gestionar datos de solicitud de servicios de un usuario individual

Prueba de aceptación
HU: Gestionar datos de solicitud de servicios de un usuario individual
Nombre: Prueba para comprobar la entrada de información de solicitudes de servicios de un usuario individual.
Descripción: Introducción correcta de la información de solicitudes de servicios de usuario individual.
Condiciones de ejecución: El usuario debe introducir los datos necesarios de la solicitud de servicios de usuario individual.
Entrada / Pasos ejecución: El usuario intente introducir la información de la solicitud.
Resultado: Se emite un mensaje de error en caso de: • No se seleccionen servicios. • Se inserte un carné de identidad que ya existe. • Existan campos vacíos en el formulario. • Se inserte un nombre de usuario que ya existe. • En caso contrario se emite un mensaje de éxito en la inserción.
Evaluación de la prueba: Aceptada.

Pruebas de aceptación para la HU: Gestionar datos de solicitud de servicios de los Estudiantes

Prueba de aceptación
HU: Gestionar datos de solicitud de servicios de los Estudiantes
Nombre: Prueba para comprobar la entrada de información de solicitudes de servicios de estudiantes.
Descripción: Validación de la entrada correcta de información de las solicitudes de servicios de estudiantes.
Condiciones de ejecución: El usuario debe introducir los datos necesarios de la solicitud de servicios de estudiantes.
Entrada/Pasos ejecución: El usuario intenta introducir la información de la solicitud de estudiante.
Resultado: Se emite un mensaje de error en caso de: • Se inserte una solicitud de un grupo que ya existe. • Se inserte un carné de identidad de un estudiante que ya existe. • Existan campos vacíos en el formulario. • Se inserte un nombre de usuario que ya existe. • En caso contrario se emite un mensaje de éxito en la inserción.
Evaluación de la prueba: Aceptada.

Prueba de aceptación para la HU: Gestionar autorizo por el decano

Prueba de aceptación
HU: Gestionar autorizo por el decano.
Nombre: Prueba para comprobar el autorizo por los decanos
Descripción: Validación de la selección de los servicios que se desea autorizar.
Condiciones de ejecución: El usuario debe seleccionar los servicios que él desea autorizar.
Entrada/Pasos ejecución: El usuario intenta guardar el autorizo.
Resultado: Se emite un mensaje de error en caso de: Que el usuario intente guardar el autorizo y no se hallan seleccionado los servicios a autorizar.
Evaluación de la prueba: Aceptada

Prueba de aceptación para la HU: Gestionar autorizo por el V.R.I.P

Prueba de aceptación
HU: Gestionar autorizo por el V.R.I.P.
Nombre: Prueba para comprobar el autorizo por el V.R.I.P.
Descripción: Validación de la selección de los servicios que se desea autorizar.
Condiciones de ejecución: El usuario debe seleccionar los servicios que él desea autorizar.
Entrada/Pasos ejecución: El usuario intenta guardar el autorizo.
Resultado: Se emite un mensaje de error en caso de: • Que el usuario intente guardar el autorizo y no se hallan seleccionado los servicios a autorizar.
Evaluación de la prueba: Aceptada

Prueba de aceptación para la HU: Gestionar bajas de utilización de servicios

Prueba de aceptación
HU: Gestionar bajas de utilización de servicios.
Nombre: Prueba para comprobar el informe de las bajas.
Descripción: Validación de la selección de los usuarios bajas.
Condiciones de ejecución: El usuario debe seleccionar los usuarios a dar baja.
Entrada/Pasos ejecución: El usuario intenta informar bajas de usuarios.
Resultado: Se emite un mensaje de error en caso de: • Que el usuario intente informar las bajas y no se hallan seleccionado usuarios a dar baja.
Evaluación de la prueba: Aceptada



Prueba de aceptación para la HU: Saber estado de la solicitud

Prueba de aceptación
HU: Saber estado de la solicitud.
Nombre: Prueba para comprobar la búsqueda del estado de las solicitudes individuales.
Descripción: Validación de la entrada del carné de identidad del solicitante que busca el estado de su solicitud.
Condiciones de ejecución: El usuario de entrar el carné de identidad.
Entrada/Pasos ejecución: El usuario intenta buscar el estado de su solicitud.
Resultado: Se emite un mensaje de error en caso de: El carné insertado no sea de ningún solicitante del sistema.
Evaluación de la prueba: Aceptada

ANEXO 8: INTERFASES DE USUARIO



Fig-1 Página principal del sistema.



Fig-2 Configurar sistema.



SAGASSI
Sistema Automatizado para la Gestión y Aprobación de Solicitudes de Servicios Informáticos

Inicio | Estado de: Admin | Desconectar | Cerrar Sesión | Salir

Formulario de solicitud de un Usuario Individual

Nombre:

Apellido:

Carné de Identidad:

Dirección Particular:

Categoría:

Código de Identificación de entrada:

Cargo:

Nombre de Usuario (Correo, Identificación):

Contraseña para la solicitud de servicio:

Tipo de servicio a la cual (debe de llevar el prefijo de la categoría en cada servicio):

☐ Acceso Correo Institucional

☐ Acceso Internet

☐ Acceso al Jabber

☐ Acceso a la Red ISMM

Fig-3 Formulario de solicitud Individual.



SAGASSI
Sistema Automatizado para la Gestión y Aprobación de Solicitudes de Servicios Informáticos

Inicio | Estado de: Decano | Cerrar Sesión | Salir

Autorizo por los Decanos

Departamento:

☐ Mostrar Solicitudes de Estudiantes

☐ Mostrar Solicitudes de Usuario Individual

Lista de Usuarios con sus Servicios

Usuario	Servicios	Id	Id
informatica2010	Acceso Correo Nacional, Acceso al Jabber, Acceso a la Red ISMM		
kenwiko	kenwiko	382308	informatica[10]
wokid7	kenwiko	32848238	informatica[10]

Fig-4 Autorizo por el decano.

SAGASSI
Sistema Automatizado para la Gestión y Aprobación de Solicitudes de Servicios Informáticos

Inicio | Usuario: V.R.I.P. | Cerrar Sesión afortunati

Autorizo por el V.R.I.P.

Facultades del Centro:
metalurgia-electromecanica
Departamento de la Facultad:
Eléctrica

☐ Mostrar Solicitudes de Estudiantes
☐ Mostrar Solicitudes de Usuario Individual

Unidad de Usuario con sus Datos:

Nombre	Apellido	Nombre de Usuario	Apellido	Acciones autorizadas
yastray	ola vez cruz	a chidiasle	jwehli	☒ Acceso Correo Internacional ☒ Acceso a Internet ☐ Acceso al Jabbber
ya anin	ola vez cruz	veglew	sqhej	☒ Acceso Correo Internacional ☒ Acceso a Internet ☐ Acceso al Jabbber
alvarado	torres cruz	sqhejhi	keanqjeahli	☐ Acceso a Internet ☒ Acceso al Jabbber ☒ Acceso a la Red ISMBA

Guardar Autorizo

Fig-5 Autorizo por el V.R.I.P.