

**UNIVERSIDAD DE LAS CIENCIAS
INFORMÁTICAS
FACULTAD 8**



**“DESARROLLO DEL SITIO WEB PARA LOS
PROFESORES GUÍAS DE LA FACULTAD 8”**

Autor:

Danais Rojas Suárez.

Tutor:

Ing. Arcel Labrada Batista

Co-tutor:

Ing. Héctor Pérez Hernández

Ciudad de La Habana, febrero 2010

“Año del 51 aniversario de la Revolución”

DECLARACIÓN DE AUTORÍA

Declaro que soy la única autora del trabajo de diploma: Desarrollo del Sitio Web para los profesores guías de la Facultad 8 y autorizo a la Facultad 8 de la Universidad de las Ciencias Informáticas a hacer uso del mismo en su beneficio.

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

Autora:

Danais Rojas Suárez

Tutor:

Ing. Arcel Labrada Batista

Co-Tutor:

Ing. Héctor Pérez Hernández

DEDICATORIA

... A mis padres por todo el apoyo, la confianza y el amor que siempre me han dado, por haberme entregado sus vidas sin esperar nada a cambio. Sin su ejemplo no hubiera podido lograr lo que he sido hasta hoy.

AGRADECIMIENTOS

A mis padres por confiar siempre en mí, por estar a mi lado en cada momento de mi vida, por sus enseñanzas, sus consejos, su comprensión y respetar mis decisiones, por su sacrificio y su infinito amor, por haber creído siempre que podría llegar a ser quien soy hoy. En fin porque son la guía de mi vida y la mayor razón de mi existencia.

A mi hermano por su apoyo y su preocupación.

A mi familia y todos aquellos que se le han sumado, por cada muestra de amor, confianza y por cada palabra de orgullo.

A mi novio que en estos meses me ha apoyado incondicionalmente en todo y por haber estado a mi lado cuando más lo he necesitado.

A todas mis amigas y amigos por permitirme contar con la dicha de tenerlos y hacer de estos cinco años juntos una parte de mi vida que jamás olvidaré.

Y sobre todo a Denis por haber sido parte fundamental en la realización de este trabajo.

RESUMEN

Este trabajo contiene la investigación y desarrollo de una aplicación web que gestione toda la información referente a los estudiantes de la Facultad 8 de la Universidad de las Ciencias Informáticas. El mismo pretende facilitar la labor de los profesores guías, obteniendo un mejor desempeño de estos. El trabajo incluye un estudio del estado del arte, las herramientas utilizadas, características del sistema, planificación, implementación y la estrategia de pruebas que se llevó a cabo. Además un conjunto de conclusiones y recomendaciones que consideramos de gran importancia para el tema tratado.

ÍNDICE DE CONTENIDO

INTRODUCCIÓN.....	13
CAPÍTULO 1.....	16
1.1 INTRODUCCIÓN.....	16
1.2 ACTUALIDAD DEL TRABAJO.....	16
1.3 TECNOLOGÍAS A UTILIZAR.....	17
1.3.1 Tipo de aplicación.....	17
1.3.2 Gestores de Base Datos.....	17
1.3.3 Tecnologías de Desarrollo Web.....	19
1.3.4 Diferentes sistemas de gestión de contenido.....	21
1.3.5 Principales lenguajes de programación.....	26
1.4 HERRAMIENTAS UTILIZADAS.....	29
1.4.1 Herramientas de modelado.....	29
Visual Paradigm.....	30
1.4.2 Entorno de desarrollo Integrado.....	31
Zend Studio.....	31
1.5 SERVIDORES WEB.....	32
1.5.1 Internet Information Server.....	32
1.5.2 Apache.....	32
1.5.3 ¿Por qué usar Apache?.....	33
1.6 METODOLOGÍA ÁGIL.....	33
1.6.1 Principios Del manifiesto ágil.....	33
1.6.2 Selección de la Metodología.....	40
1.7 CONCLUSIONES.....	40
CAPÍTULO 2.....	41
2.1 INTRODUCCIÓN.....	41
2.2 DESCRIPCIÓN DE LAS ACCIONES VINCULADAS AL CAMPO DE ACCIÓN.....	41
2.2.1 Objetos de automatización.....	41
2.2.2 Propuesta del sistema.....	42
2.3 FASE DE EXPLORACIÓN.....	42
2.4 FASE DE PLANIFICACIÓN.....	48
2.4.1 Estimación de esfuerzos por historias de usuario.....	49
2.4.2 Plan de iteraciones.....	50
2.4.3 Plan de duración de las iteraciones.....	50
2.4.4 Plan de entregas.....	52
2.5 CONCLUSIONES.....	54
CAPÍTULO 3.....	55
3.1 INTRODUCCIÓN.....	55
3.2 DISEÑO DEL SISTEMA.....	55
3.2.1 Módulos de Drupal.....	55
3.3 DISEÑO DE LA BASE DE DATOS.....	58
3.4 FASE DE IMPLEMENTACIÓN.....	60
3.4.1 1ra Iteración.....	60

3.4.2 2da Iteración	62
3.4.3 3ra Iteración.....	64
3.4.4 Diagrama de Despliegue.....	66
3.5 PRUEBAS	66
3.5.1 Pruebas de aceptación.....	67
3.6 CONCLUSIONES	69
CONCLUSIONES	70
REFERENCIAS BIBLIOGRÁFICAS	72
BIBLIOGRAFÍA	74
GLOSARIO DE TÉRMINOS	76
ANEXOS.....	77

ÍNDICE DE TABLAS

TABLA 2.1 PLANTILLA DE HISTORIA DE USUARIO	43
TABLA 2.2 HU GESTIONAR CUENTA DE USUARIO.	43
TABLA 2.3 HU AUTENTICAR.....	44
TABLA 2.4 HU GESTIONAR PROYECTO EDUCATIVO.....	44
TABLA 2.5 HU CONSULTAR PROYECTO EDUCATIVO.....	44
TABLA 2.6 HU GESTIONAR GRUPOS	45
TABLA 2.7 HU GESTIONAR DESCARGAS.	45
TABLA 2.8 HU GESTIONAR AÑOS.....	46
TABLA 2.9 HU GESTIONAR CURSO.	46
TABLA 2.10 HU GESTIONAR DEPARTAMENTO.	46
TABLA 2.11 HU CONSULTAR DEPARTAMENTO.	47
TABLA 2.12 HU GESTIONAR PROCEDENCIA.....	47
TABLA 2.13 HU CONSULTAR PROCEDENCIA.....	48
TABLA 2.14 HU CONSULTAR PERFIL DE USUARIO.....	77
TABLA 2.15 HU CONSULTAR GRUPOS.....	77
TABLA 2.16 HU CONSULTAR AÑOS.....	78
TABLA 2.17 HU CONSULTAR CURSO.....	78
TABLA 2.18 HU GESTIONAR PERSONA.....	78
TABLA 2.19 HU CONSULTAR PERSONA.....	79
TABLA 2.20 HU VINCULAR GRUPO A UN CURSO EN UN SEMESTRE.....	79
TABLA 2.21 HU VINCULAR PROFESOR GUÍA A UN GRUPO.....	80
TABLA 2.22 HU VINCULAR ESTUDIANTES AL GRUPO.....	80
TABLA 2.23 HU GESTIONAR VÍNCULOS.....	80
TABLA 2.24 HU CONSULTAR VÍNCULOS.....	81
TABLA 2.25 HU GESTIONAR AVISOS.....	81
TABLA 2.26 HU CONSULTAR AVISOS.....	81
TABLA 2.27 HU GESTIONAR NOTICIAS.....	82
TABLA 2.28 HU CONSULTAR NOTICIAS.....	82
TABLA 2.29 ESTIMACIÓN DE ESFUERZOS POR HISTORIA DE USUARIO.....	49
TABLA 2.30 PLAN DE DURACIÓN DE LAS ITERACIONES.....	51
TABLA 2.31 MÓDULOS Y HISTORIAS DE USUARIO ABARCADAS.....	52

TABLA 2.32 PLAN DE DURACIÓN DE ENTREGA.	54
TABLA 3.1 TARJETA CRC MÓDULO CURSO.	56
TABLA 3.2 TARJETA CRC MÓDULO AVISO.	56
TABLA 3.3 TARJETA CRC MÓDULO NOTICIAS.	56
TABLA 3.4 TARJETA CRC MÓDULO DESCARGAS.	56
TABLA 3.5 TARJETA CRC MÓDULO VÍNCULOS.	57
TABLA 3.6 TARJETA CRC MÓDULO AÑO	57
TABLA 3.7 TARJETA CRC MÓDULO DEPARTAMENTO.	57
TABLA 3.8 TARJETA CRC MÓDULO PROCEDENCIA	57
TABLA 3.9 TARJETA CRC MÓDULO PROYECTO EDUCATIVO	58
TABLA 3.10 TARJETA CRC MÓDULO PERSONA.	58
TABLA 3.11 TARJETA CRC MÓDULO GRUPO	58
TABLA 3.12 MÓDULOS TRATADOS EN LA PRIMERA ITERACIÓN.	60
TABLA 3.13 TAREA 1 DEL MÓDULO PERSONA.	61
TABLA 3.14 TAREA 2 DEL MÓDULO PERSONA.	61
TABLA 3.15 TAREA 3 DEL MÓDULO PERSONA.	62
TABLA 3.16 TAREA 1 DEL MÓDULO PROCEDENCIA.	83
TABLA 3.17 TAREA 2 DEL MÓDULO PROCEDENCIA.	83
TABLA 3.18 TAREA 1 DEL MÓDULO GRUPO.	83
TABLA 3.19 TAREA 1 DEL MÓDULO PROYECTO_ EDUCATIVO.	84
TABLA 3.20 TAREA 2 DEL MÓDULO PROYECTO_ EDUCATIVO.	84
TABLA 3.21 TAREA 3 DEL MÓDULO PROYECTO_ EDUCATIVO.	84
TABLA 3.22 TAREA 1 DEL MÓDULO AÑO.	85
TABLA 3.23 TAREA 1 DEL MÓDULO DESCARGA.	85
TABLA 3.24 TAREA 2 DEL MÓDULO DESCARGA.	85
TABLA 3.25 TAREA 1 DEL MÓDULO CURSO.	86
TABLA 3.26 TAREA 1 DEL MÓDULO DEPARTAMENTO.	86
TABLA 3.27 TAREA 2 DEL MÓDULO DEPARTAMENTO.	87
TABLA 3.28 TAREA 3 DEL MÓDULO DEPARTAMENTO.	87
TABLA 3.29 MÓDULOS TRATADOS EN LA SEGUNDA ITERACIÓN.	62
TABLA 3.30 TAREA 2 DEL MÓDULO GRUPO.	63
TABLA 3.31 TAREA 3 DEL MÓDULO GRUPO.	63
TABLA 3.32 TAREA 4 DEL MÓDULO GRUPO.	63
TABLA 3.33 TAREA 5 DEL MÓDULO GRUPO.	64
TABLA 3.34 TAREA 4 DEL MÓDULO PERSONA.	87

TABLA 3.35 TAREA 5 DEL MÓDULO PERSONA.	88
TABLA 3.36 TAREA 2 DEL MÓDULO CURSO.	88
TABLA 3.37 TAREA 2 DEL MÓDULO AÑO.	88
TABLA 3.38 MÓDULOS TRATADOS EN LA TERCERA ITERACIÓN	64
TABLA 3.39 TAREA 1 DEL MÓDULO VÍNCULOS.	65
TABLA 3.40 TAREA 2 DEL MÓDULO VÍNCULOS.	65
TABLA 3.41 TAREA 1 DEL MÓDULO NOTICIAS.	89
TABLA 3.42 TAREA 2 DEL MÓDULO NOTICIAS.	89
TABLA 3.43 TAREA 1 DEL MÓDULO AVISOS.	90
TABLA 3.44 TAREA 2 DEL MÓDULO AVISOS.	90
TABLA 3.45 TAREA 6 DEL MÓDULO PERSONA.	90
TABLA 3.46 PRUEBA1 AL MÓDULO PERSONA.	67
TABLA 3.47 PRUEBA 2 AL MÓDULO PERSONA.	68
TABLA 3.48 PRUEBA 3 AL MÓDULO PERSONA.	68
TABLA 3.49 PRUEBA 1 AL MÓDULO CURSO.	91
TABLA 3.50 PRUEBA 2 AL MÓDULO CURSO.	91
TABLA 3.51 PRUEBA 1 AL MÓDULO GRUPO.	92
TABLA 3.52 PRUEBA 2 AL MÓDULO GRUPO.	92
TABLA 3.53 PRUEBA 3 AL MÓDULO GRUPO.	93
TABLA 3.54 PRUEBA 4 AL MÓDULO GRUPO.	93
TABLA 3.55 PRUEBA 5 AL MÓDULO GRUPO.	94
TABLA 3.56 PRUEBA 1 AL MÓDULO AVISOS.	94
TABLA 3.57 PRUEBA 2 AL MÓDULO AVISOS.	95
TABLA 3.58 PRUEBA 1 AL MÓDULO NOTICIAS.	95
TABLA 3.59 PRUEBA 2 AL MÓDULO NOTICIAS.	95
TABLA 3.60 PRUEBA 1 AL MÓDULO DESCARGAS.	96
TABLA 3.61 PRUEBA 1 AL MÓDULO VÍNCULO.	96
TABLA 3.62 PRUEBA 2 AL MÓDULO VÍNCULO.	97
TABLA 3.63 PRUEBA 1 AL MÓDULO AÑO.	97
TABLA 3.64 PRUEBA 2 AL MÓDULO AÑO.	97
TABLA 3.65 PRUEBA 1 AL MÓDULO DEPARTAMENTO.	98
TABLA 3.66 PRUEBA 2 AL MÓDULO DEPARTAMENTO.	98
TABLA 3.67 PRUEBA 1 AL MÓDULO PROCEDENCIA.	99
TABLA 3.68 PRUEBA 2 AL MÓDULO PROCEDENCIA.	99
TABLA 3.69 PRUEBA 1 AL MÓDULO PROYECTO EDUCATIVO.	99

TABLA 3.70 PRUEBA 2 AL MÓDULO PROYECTO EDUCATIVO.	100
---	------------

ÍNDICE DE FIGURAS

<i>FIGURA 1 MODELO DE DATOS</i>	59
<i>FIGURA 2 DIAGRAMA DE DESPLIEGUE</i>	66

INTRODUCCIÓN

Las universidades juegan un papel crucial en la transformación y el progreso de las comunidades actuales a nivel mundial. Estas constituyen una importante fuente de profesionales, que en un futuro formarán los pilares de la sociedad. Es por ello que la formación de los estudiantes es tarea prioritizada de cualquier país en desarrollo, porque de ellos es de donde se nutrirá la nueva generación de profesionales. La Revolución Cubana ha dado gran prioridad a las casas de altos estudios, haciendo bastos esfuerzos para que sus graduados tengan no solo una buena preparación científica, sino que también cuenten con un alto valor ético y moral; de ahí se deriva la importancia de la labor de los profesores en general, y dentro de estos muy especialmente la del profesor guía como orientador del grupo que se le asigna. Es por esto que se realizó una revisión del tema destacando algunos elementos que se consideran de gran importancia, detectándose algunos problemas como la falta de experiencia existente en el rol de profesores guías en la Facultad 8 de la Universidad de las Ciencias Informáticas, generada por la poca información que poseen sobre las funciones que deben cumplir y las características generales del trabajo, así como por la juventud del claustro, impidiendo lograr la excelencia en esta área. Estos profesores necesitan conocer mucha información de sus estudiantes ya que realizan un proyecto educativo para sus brigadas donde trazan las metas que el grupo debe cumplir en el transcurso del año. En este se recoge información referente a la docencia, la extensión, y de carácter político y personal. Al no existir un control adecuado de esta información generada en el proceso de tutoría de los grupos docentes, propicia que los datos se pierdan con gran facilidad en la mayoría de los casos y no permite hacer la entrega pedagógica, proceso natural que queda trunco en la facultad. A lo que se le suma la no existencia de mecanismos coherentes de control por parte del aparato administrativo del trabajo realizado por los profesores guías, lo que impide la evaluación eficiente del trabajo de dichos profesores.

Por lo antes expuesto, el **problema a solucionar** sería: ¿Cómo gestionar toda la información de interés que recogen los profesores guías referentes a sus estudiantes en la facultad 8 de la Universidad de las Ciencias Informáticas? Como **objetivo**

general se propone desarrollar un sistema que permita gestionar la información que los profesores guías manejan de sus grupos. El **objeto de estudio** es el proceso de gestión de información. Como **objetivos específicos** se tienen:

1. Realizar el estudio sobre enfoques teóricos.
2. Definir el lenguaje, metodología y herramientas de desarrollo de software que permita gestionar el proceso relacionado a los profesores guías.
3. Desarrollar el sistema siguiendo la metodología seleccionada.
4. Realizar pruebas de aceptación al sistema.

El **campo de acción** es la gestión de la información de interés de los profesores guías sobre sus estudiantes de la Facultad 8. Como **idea a defender** se plantea que si se implementa un sistema que informatice la gestión de la información de interés sobre los estudiantes entonces se permitirá un mejor trabajo y una mayor organización de la labor del profesor guía.

En este trabajo se emplearan métodos de investigación científica como:

Métodos Teóricos:

- Histórico – Lógico

Con este método se realizará una investigación con el objetivo de conocer el desarrollo de otras aplicaciones Web para la gestión de información docente desde que comenzaron a utilizarse con este fin.

- Analítico – Sintético

Se utilizará para analizar las teorías y documentos relacionados con la labor de los profesores guías en la Facultad 8 de la Universidad de las Ciencias Informáticas, para extraer los elementos más importantes que se relacionan con esta labor educativa.

- Modelación

Se utilizará este método porque permite crear un prototipo de la aplicación que permite la representación de todos los requerimientos que debe cumplir el sistema.

Métodos Empíricos:

- Observación

Se usará para obtener un resultado visual haciendo exámenes reales al software para determinar que cumpla con los objetivos y detectar errores si es que existen.

- Experimento

Mediante este método se comprobará que la aplicación funcione de manera eficiente y cumpla los objetivos para los cuales fue creada, mediante pruebas al sistema utilizando datos artificiales.

CAPÍTULO 1

Fundamentación Teórica

1.1 Introducción.

En este capítulo se desarrolla un análisis profundo del problema y las diferentes tecnologías a utilizar en el desarrollo del sistema.

1.2 Actualidad del Trabajo

En la actualidad en Cuba no existe ninguna referencia de que exista en alguna universidad algún sistema de gestión para los profesores guías, donde se gestione toda la información referente a la labor que ellos cumplen. También se realizó una búsqueda a nivel internacional y no se logró encontrar ningún portal que gestionara este tipo de información de la manera que la facultad requiere. Se encontraron sistemas que si gestionan información sobre los estudiantes como Akademos (en la Universidad de las ciencias informáticas) que brinda una misma interfaz para todos los usuarios y muestra la información de formación de cada estudiante, pero no cumple con los requerimientos del sistema que se pretende implementar, por ejemplo no permite la gestión de los proyectos educativos de año, de la facultad y de las brigadas, tampoco cuenta con la opción de poder brindarle a los profesores la posibilidad de ver en cuantos grupos han estado sus estudiantes, así como cuales han sido sus evaluaciones en los diferentes semestres por los que ha pasado y las observaciones que se le han hecho en cada una de estas evaluaciones. De forma similar se encontró el Sistema de Gestión de la Nueva Universidad (SIGENU) que es un software que soporta el almacenamiento de información agregada y nominalizada de cualquier estudiante que pertenezca a una Institución de Educación Superior de Cuba, presentada en la Feria Internacional Informática 2009 por el Instituto Superior Politécnico José Antonio Echeverría y adaptable a primera vista a nuestras necesidades, pero que igualmente no cumple todos los requisitos necesarios para la

gestión de la información que recogen los profesores guías en la facultad 8 de la UCI mencionados anteriormente.

1.3 Tecnologías a utilizar.

1.3.1 Tipo de aplicación.

Las aplicaciones Web son aplicaciones que los usuarios pueden utilizar accediendo a un servidor Web a través de Internet o de una intranet mediante un navegador. Estas son populares debido a la facilidad para actualizar y mantenerlas sin tener que distribuir e instalar software a miles de usuarios. Tienen numerosas ventajas en las que se encuentran:

- Se puede migrar de sistema operativo o cambiar el Hardware libremente sin afectar el funcionamiento de las aplicaciones de servidor.
- No se requieren complicadas combinaciones de Hardware/Software para utilizar estas aplicaciones. Solo un computador con un buen navegador Web.
- Se facilita el trabajo a distancia. Se puede trabajar desde cualquier PC o computador portátil con conexión a Internet. Actualizar o hacer cambios en el Software es sencillo y sin riesgos de incompatibilidades.
- Existe solo una versión en el servidor lo que implica que no hay que distribuirla entre los demás computadores. El proceso es rápido y limpio.
- Pueden ser utilizadas por varios usuarios al mismo tiempo.

1.3.2 Gestores de Base Datos.

Un sistema gestor de base de datos se define como el conjunto de programas que administran y gestionan la información contenida en una base de datos. Existen varios gestores de base datos, entre los cuales se encuentran:

MySQL.

Este gestor de bases de datos es, probablemente, el gestor más usado en el mundo, debido a su gran rapidez y facilidad de uso. Esta gran aceptación es debida, en parte, a que existen infinidad de librerías y otras herramientas que permiten su uso a través

de gran cantidad de lenguajes de programación, además de su fácil instalación y configuración.

Las principales características de este gestor de bases de datos son las siguientes:

Interioridades y portabilidad.

- Escrito en C y en C++.
- Funciona en diferentes plataformas.
- APIs disponibles para C, C++, PHP, Python y Ruby,

Seguridad.

- Un sistema de privilegios y contraseñas que es muy flexible y seguro, y que permite verificación basada en el host. Las contraseñas son seguras porque todo el tráfico de contraseñas está encriptado cuando se conecta con un servidor.

Conectividad.

- Los clientes pueden conectar con el servidor MySQL usando sockets TCP/IP en cualquier plataforma. En sistemas Windows de la familia NT (NT, 2000, XP, o 2003). En sistemas Unix, los clientes pueden conectar usando ficheros socket Unix. (ALBERTO 2010)

PostgreSQL.

PostgreSQL está considerado como la base de datos de código abierto más avanzada del mundo. A continuación se enumeran las principales características de este gestor de bases de datos:

- Soporta distintos tipos de datos: además del soporte para los tipos base, también soporta datos de tipo fecha, monetarios, elementos gráficos, datos sobre redes (MAC, IP...), cadenas de bits, etc. También permite la creación de tipos propios.
- Incorpora una estructura de datos en arreglos.

- Incorpora funciones de diversa índole: manejo de fechas, geométricas, orientadas a operaciones con redes, etc.
- Permite la declaración de funciones propias, así como la definición de disparadores.
- Soporta el uso de índices, reglas y vistas.
- Incluye herencia entre tablas (aunque no entre objetos, ya que no existen), por lo que a este gestor de bases de datos se le incluye entre los gestores objeto-relacionales.
- Permite la gestión de diferentes usuarios, como también los permisos asignados a cada uno de ellos. (DANIEL)

Selección del Sistema Gestor de Bases de Datos (SGBD) a utilizar.

Después de analizar las características de algunos sistemas gestores de base de datos se llega a la conclusión de que aunque MySQL es el más popular de todos los SGBD por su gran velocidad en las operaciones, su bajo consumo y su facilidad de administración, no es el más indicado para el sistema que se desarrolla. MySQL carece de soporte para transacciones, rollback's y subconsultas. Además no maneja la integridad referencial y toda la documentación que se puede encontrar es propietaria.

Por tanto se elige PostgreSQL como el gestor que se utilizará en el desarrollo del portal, ya que este posee una gran escalabilidad, pues es capaz de ajustarse el número de CPUs y a la cantidad de memoria que posee el sistema de forma óptima, poseyendo soporte para una mayor cantidad de peticiones simultáneas de manera correcta. Además tiene la capacidad de comprobar la integridad referencial, así como también la de almacenar procedimientos en la propia base de datos, equiparándolo con los gestores de bases de datos de alto nivel.

1.3.3 Tecnologías de Desarrollo Web.

Framework.

Un framework es un esquema (un esqueleto, un patrón) para el desarrollo o la implementación de una aplicación. (SÁNCHEZ 2007) Su utilización tiene gran ventaja, pues el programador no necesita plantearse una estructura global de la aplicación, sino

que el framework le proporciona un esqueleto que hay que "rellenar". También facilita la colaboración, pues todo lo que sea definir y estandarizar va a ahorrar tiempo y trabajo a los desarrollos en grupo. Y por otra parte es más fácil encontrar herramientas (utilidades, librerías) adaptadas al framework concreto para facilitar el desarrollo.

Sistema de gestión de contenidos.

Un Sistema de Gestión de Contenidos (CMS) es una aplicación que sirve para simplificar el proceso de crear, publicar y gestionar los contenidos de un sitio Web.

Los Sistemas de Gestión de Contenidos permiten renovar los contenidos de una Web de una forma muy fácil a usuarios que desconozcan el lenguaje XHTML. También hacen posible de forma extremadamente sencilla la carga y descarga de archivos de texto o imagen. Estos cambios en los contenidos se realizan accediendo a una página en la que tras introducir un nombre de usuario y una contraseña se encontrará con un menú de opciones muy sencillo e intuitivo (Insertar Modificar, Borrar...) a través del cual podrá cambiar los contenidos de su Web en cuestión de segundos.

Características de los Sistemas Gestión de Contenidos.

Los Sistemas Gestión de Contenidos ofrecen la posibilidad de realizar las siguientes funciones de forma extremadamente fácil:

- Añadir nuevos contenidos.
- Editar contenidos.
- Borrar contenidos antiguos.
- Mantener una sección de Noticias.
- Subir imágenes al servidor.
- Cargar archivos en el servidor (PDFs, DOCs...) y ponerlos a disposición de los usuarios para su descarga.
- Insertar enlaces hacia otras páginas.
- El sistema puede ser mantenido por varias personas que pueden acceder de forma independiente al Administrador de Contenidos. (ATRIOWEB)

1.3.4 Diferentes sistemas de gestión de contenido.

Joomla.

Joomla es un Sistema de Gestión de Contenidos (CMS) reconocido mundialmente, que le ayuda a construir sitios Web y otras aplicaciones en línea potentes. Lo mejor de todo, es que Joomla es una solución de código abierto y está disponible libremente para cualquiera que desee utilizarlo.

- **Organización del sitio Web:** Joomla está preparado para organizar eficientemente los contenidos de su sitio en secciones y categorías, lo que facilita la navegabilidad para los usuarios y permite crear una estructura sólida, ordenada y sencilla para los administradores. Desde el panel administrador de Joomla usted podrá crear, editar y borrar las secciones y categorías de su sitio de la manera en que más le convenga.
- **Publicación de Contenidos:** Con Joomla CMS podrá crear páginas ilimitadas y editarlas desde un sencillo editor que permite formatear los textos con los estilos e imágenes deseados. Los contenidos son totalmente editables y modificables.
- **Escalabilidad e implementación de nuevas funcionalidades:** Joomla ofrece la posibilidad de instalar, desinstalar y administrar componentes y módulos, que agregarán servicios de valor a los visitantes de su sitio Web, por ejemplo: galerías de imágenes, foros, clasificados, etc.
- **Administración de usuarios:** Joomla le permite almacenar datos de usuarios registrados y también la posibilidad de enviar E-mails masivos a todos los usuarios. La administración de usuarios es jerárquica, y los distintos grupos de usuarios poseen diferentes niveles de facultades/permisos dentro de la gestión y administración del sitio.
- **Diseño y aspecto estético del sitio:** Es posible cambiar todo el aspecto del sitio Web tan solo con un par de clicks, gracias al sistema de plantillas que utiliza Joomla.
- **Navegación y menú:** Totalmente editables desde el panel administrador de Joomla

- **Administrador de Imágenes:** Joomla posee una utilidad para subir imágenes al servidor y usarlas en todo el sitio.
- **Disposición de módulos modificable:** En un sitio creado con Joomla, la posición de módulos puede acomodarse como se prefiera.
- **Estadísticas de visitas:** con información de navegador y detalles de los documentos (páginas) más vistos.

Características de publicación de páginas Web en Joomla.

- **Automatización en la publicación:** Las páginas y documentos de Joomla pueden programarse con fecha de publicación y fecha de caducidad. Es decir un documento puede programarse para que se publique automáticamente al llegar una determinada fecha, y luego termine la publicación también de forma automática en otra fecha.
- **Archivo e historial:** Las páginas viejas o publicaciones que hayan perdido vigencia pueden enviarse a un "archivo" de almacenamiento, sin necesidad de tener que borrarlas. Esto permite también dar la posibilidad a los navegantes de consultar artículos viejos o documentos anteriores en un historial.
- **Formatos de lectura:** Cada documento es generado automáticamente por Joomla en formato PDF, en versión imprimible, y en XML.
- **Valoración de contenidos:** Los visitantes del sitio podrán votar la calidad de lo publicado.
- **Comentarios:** (opcional) Los usuarios podrán comentar sus opiniones o expresar sus inquietudes en la misma página de contenidos. (CERONJC 2009)

Drupal.

Características generales.

Ayuda on-line: Un robusto sistema de ayuda online y páginas de ayuda para los módulos del 'núcleo', tanto para usuarios como para administradores.

Búsqueda: Todo el contenido en Drupal es totalmente indexado en tiempo real y se puede consultar en cualquier momento.

Código abierto: El código fuente de Drupal está libremente disponible bajo los términos de la licencia GNU/GPL. Al contrario que otros sistemas de 'blogs' o de gestión de contenido propietarios, es posible extender o adaptar Drupal según las necesidades.

Módulos: La comunidad de Drupal ha contribuido muchos módulos que proporcionan funcionalidades como 'página de categorías', autenticación mediante jabber, mensajes privados, etc.

Personalización: Un robusto entorno de personalización está implementado en el núcleo de Drupal. Tanto el contenido como la presentación pueden ser individualizados de acuerdo las preferencias definidas por el usuario.

URLs amigables: Drupal usa el mod_rewrite de Apache para crear URLs que son manejables por los usuarios y los motores de búsqueda.

Gestión de usuarios.

Autenticación de usuarios: Los usuarios se pueden registrar e iniciar sesión de forma local o utilizando un sistema de autenticación externo.

Permisos basados en roles: Los administradores de Drupal no tienen que establecer permisos para cada usuario. En lugar de eso, pueden asignar permisos a un 'rol' y agrupar los usuarios por roles.

Gestión de contenido.

Control de versiones: El sistema de control de versiones de Drupal permite seguir y auditar totalmente las sucesivas actualizaciones del contenido: qué se ha cambiado, la hora y la fecha, quién lo ha cambiado, y más. También permite mantener comentarios sobre los sucesivos cambios o deshacer los cambios recuperando una versión anterior.

Enlaces permanentes (Permalinks): Todo el contenido creado en Drupal tiene un enlace permanente asociado a él para que pueda ser enlazado externamente sin temor de que el enlace falle en el futuro.

Objetos de Contenido (Nodos): El contenido creado en Drupal es, funcionalmente, un objeto (Nodo). Esto permite un tratamiento uniforme de la información, como una misma cola de moderación para envíos de diferentes tipos, promocionar cualquiera de estos objetos a la página principal o permitir comentarios -o no- sobre cada objeto.

Plantillas (Templates): El sistema de temas de Drupal separa el contenido de la presentación permitiendo controlar o cambiar fácilmente el aspecto del sitio Web. Se pueden crear plantillas con HTML y/o con PHP.

Bloggng.

Agregador de noticias: Drupal incluye un potente Agregador de noticias para leer y publicar enlaces a noticias de otros sitios Web. Incorpora un sistema de caché en la base de datos, con temporización configurable.

Plataforma.

Independencia de la base de datos: Aunque la mayor parte de las instalaciones de Drupal utilizan MySQL, existen otras opciones. Drupal incorpora una 'capa de abstracción de base de datos' que actualmente está implementada y mantenida para MySQL y PostgreSQL, aunque permite incorporar fácilmente soporte para otras bases de datos.

Multiplataforma: Drupal ha sido diseñado desde el principio para ser multiplataforma. Puede funcionar con Apache o Microsoft IIS como servidor Web y en sistemas como Linux, BSD, Solaris, Windows y Mac OS X. Por otro lado, al estar implementado en PHP, es totalmente portable.

Múltiples idiomas y Localización: Drupal está pensado para una audiencia internacional y proporciona opciones para crear un portal multilingüe. Todo el texto puede ser fácilmente traducido utilizando una interfaz Web, importando traducciones existentes.

Administración y Análisis.

Administración vía Web: La administración y configuración del sistema se puede realizar enteramente con un navegador y no precisa de ningún software adicional.

Análisis, Seguimiento y Estadísticas: Drupal puede mostrar en las páginas Web de administración informes sobre referrals (enlaces entrantes), popularidad del contenido, o de cómo los usuarios navegan por el sitio.

Características de comunidad.

Comentarios enlazados: Drupal proporciona un potente modelo de comentarios enlazados que posibilita seguir y participar fácilmente en la discusión sobre el comentario publicado. Los comentarios son jerárquicos, como en un grupo de noticias o un foro.

Encuestas: Drupal incluye un módulo que permite a los administradores y/o usuarios crear encuestas on-line totalmente configurables.

Foros de discusión: Drupal incorpora foros de discusión para crear sitios comunitarios vivos y dinámicos.

Libro Colaborativo: Esta característica es única de Drupal y permite crear un proyecto o "libro" a ser escrito y que otros usuarios contribuyan contenido. El contenido se organiza en páginas cómodamente navegables.

Rendimiento y escalabilidad.

Control de congestión Drupal incorpora un mecanismo de control de congestión que permite habilitar y deshabilitar determinados módulos o bloques dependiendo de la carga del servidor. Este mecanismo es totalmente configurable y ajustable.

Sistema de Caché: El mecanismo de caché elimina consultas a la base de datos incrementando el rendimiento y reduciendo la carga del servidor. (JOSE Julio 17 del 2005)

¿Por qué utilizar Drupal?

Para la realización del sistema se decidió utilizar el CMS Drupal. Tanto Drupal como Joomla son sistemas de gestión de última generación muy utilizados a nivel mundial. Dada la opinión de muchos expertos, Drupal, además de gozar de las ventajas que brinda el código abierto, brinda mucha seguridad, permitiendo a la vez una fácil administración. Además se construyen constantemente módulos que brindan infinidad de funcionalidades. Con Drupal se pueden crear encuestas y foros de manera muy

fácil y su mecanismo de caché elimina consultas a la base de datos incrementando el rendimiento y reduciendo la carga del servidor. En la Universidad de las ciencias informáticas es un sistema muy usado y que ha tenido resultados muy positivos.

1.3.5 Principales lenguajes de programación.

ASP.NET

Este es un lenguaje comercializado por Microsoft, y usado por programadores para desarrollar entre otras funciones, sitios Web. ASP.NET es el sucesor de la tecnología ASP, fue lanzada al mercado mediante una estrategia de mercado denominada .NET.

El ASP.NET fue desarrollado para resolver las limitantes que brindaba tu antecesor ASP. Creado para desarrollar Web sencillas o grandes aplicaciones. Para el desarrollo de ASP.NET se puede utilizar C#, VB.NET o J#. Los archivos cuentan con la extensión (aspx). Para su funcionamiento de las páginas se necesita tener instalado IIS con el Framework .Net.

Ventajas:

- Completamente orientado a objetos.
- Controles de usuario y personalizados.
- División entre la capa de aplicación o diseño y el código.
- Facilita el mantenimiento de grandes aplicaciones.
- Incremento de velocidad de respuesta del servidor.
- Mayor velocidad.
- Mayor seguridad.

PHP

Es un lenguaje de programación utilizado para la creación de sitio Web. PHP es un acrónimo recursivo que significa “PHP Hypertext Pre-processor”, (inicialmente se llamó Personal Home Page).

PHP es un lenguaje de script interpretado en el lado del servidor utilizado para la generación de páginas Web dinámicas, embebidas en páginas HTML y ejecutadas en el servidor. PHP no necesita ser compilado para ejecutarse. Para su funcionamiento necesita tener instalado Apache o IIS con las librerías de PHP. La mayor parte de su sintaxis ha sido tomada de C, Java y Perl con algunas características específicas. Los archivos cuentan con la extensión (php).

Ventajas:

- Incluye gran cantidad de funciones.
- Soporta en cierta medida la orientación a objeto. Clases y herencia.
- Es un lenguaje multiplataforma: Linux, Windows, entre otros.
- Se caracteriza por ser un lenguaje muy rápido.
- Capacidad de conexión con la mayoría de los manejadores de base de datos: MySQL, PostgreSQL, Oracle, MS SQL Server, entre otras.
- Capacidad de expandir su potencial utilizando módulos.
- Posee documentación en su página oficial la cual incluye descripción y ejemplos de cada una de sus funciones.
- Es libre, por lo que se presenta como una alternativa de fácil acceso para todos.
- Muy fácil de aprender.

Seguridad:

PHP es un poderoso lenguaje e intérprete, ya sea incluido como parte de un servidor Web en forma de módulo o ejecutado como un binario CGI separado, es capaz de acceder a archivos, ejecutar comandos y abrir conexiones de red en el servidor. Estas propiedades hacen que cualquier cosa que sea ejecutada en un servidor Web sea insegura por naturaleza.

JSP

Es un lenguaje para la creación de sitios Web dinámicos, acrónimo de Java Server Pages. Está orientado a desarrollar páginas Web en Java. JSP es un lenguaje multiplataforma. Creado para ejecutarse del lado del servidor.

JSP fue desarrollado por Sun Microsystems. Comparte ventajas similares a las de ASP.NET, desarrollado para la creación de aplicaciones Web potentes. Para su funcionamiento se necesita tener instalado un servidor Tomcat.

Características:

- Código separado de la lógica del programa.
- Las páginas son compiladas en la primera petición.
- Permite separar la parte dinámica de la estática en las páginas Web.
- Los archivos se encuentran con la extensión (jsp).
- El código JSP puede ser incrustado en código HTML.

Elementos de JSP

Los elementos que pueden ser insertados en las páginas JSP son los siguientes:

- Código: se puede incrustar código "Java".
- Directivas: permite controlar parámetros del servlet.
- Acciones: permite alterar el flujo normal de ejecución de una página.

Ventajas:

- Ejecución rápida del servlets.
- Crear páginas del lado del servidor.
- Multiplataforma.
- Código bien estructurado. (DAMIÁN 2007)

¿Por qué utilizar PHP?

Teniendo cuenta las características y ventajas de los lenguajes de programación Web antes expuestos se puede apreciar que todos estos lenguajes están aptos para desarrollar portales Web de óptima calidad pero en el caso de lenguaje ASP a pesar de ser de gran facilidad para los usuarios tiene los inconvenientes de presentar un código bastante desorganizado, además de tener que escribir mucho código para realizar funciones sencillas, a esto se le une la desventaja de ser tecnología propietaria. Por su parte JSP es multiplataforma y presenta un código bien estructurado, pero tiene la desventaja de ser propietario.

Por todas estas razones se escoge al lenguaje de programación PHP para llevar a cabo el desarrollo del portal Web ya que el mismo es muy fácil de aprender, se caracteriza por ser un lenguaje de programación muy rápido, es multiplataforma y tiene capacidad de conexión con la mayoría de los manejadores de bases de datos entre estos el PostgreSQL. También se puede decir que es libre, por lo que representa una alternativa de fácil acceso para todos, tiene la posibilidad de expandir su potencial mediante módulos.

1.4 Herramientas utilizadas

1.4.1 Herramientas de modelado.

Rational Rose

Es una de las más poderosas herramientas de modelado visual para el análisis y diseño de sistemas basados en objetos. Se utiliza para modelar un sistema antes de proceder a construirlo. Cubre todo el ciclo de vida de un proyecto concepción y formalización del modelo, construcción de los componentes, transición a los usuarios y certificación de las distintas fases. (SlideShare Inc, 2009)

Alguna de las ventajas que brinda son las siguientes:

- Trabajo en grupo.
- Generación de código.
- Generación de documentos automáticamente.

- Ingeniería inversa.

Esta herramienta propone la utilización de cuatro tipos de modelo para realizar un diseño del sistema, utilizando una vista estática y otra dinámica de los modelos del sistema, uno lógico y otro físico. Permite crear y refinar estas vistas creando de esta forma un modelo completo que representa el dominio del problema y el sistema de software (CARVAJAL, 2005)

Visual Paradigm

Visual Paradigm es una herramienta que sirve para realizar modelado UML siguiendo el estándar UML 2.1. Esta herramienta tiene unas características graficas muy cómodas que facilitan la realización de los diagramas de modelado que sigue el estándar de UML que son:

- Diagramas de clase, Casos de Uso, Comunicación, Secuencia, Estado, Actividad, Componentes, etc.

Entre otras características importantes que se tienen tenemos su integración con diversas IDE's como son:

- Eclipse.
- NetBeans. (ALTAMIRANO, 2009)

Algunas de sus principales características son:

- Producto de calidad.
- Licencia gratuita.
- Soporta aplicaciones Web.
- Varios idiomas.
- Generación de código para Java y exportación como HTML.
- Fácil de instalar y actualizar.
- Compatibilidad entre ediciones.

Después de haber analizado las características de las herramientas de modelado expuestas anteriormente, se concluye que ambas son muy usadas a nivel mundial y facilitan el trabajo durante el desarrollo de un producto de software garantizando su calidad final. Por tanto ambas son dos herramientas potentes y fáciles de usar. Para llevar a cabo el desarrollo del portal Web para los Profesores Guías se utilizará Visual Paradigm puesto que no hay que pagar licencias para su uso, a diferencia del Rational Rose. Además Visual Paradigm soporta aplicaciones Web.

1.4.2 Entorno de desarrollo Integrado

Zend Studio

Es un IDE propietario, compatible con las plataformas Linux, MAC y *Windows*. Incluye todos los componentes necesarios durante el ciclo de vida de una aplicación en PHP. Se puede decir que cubre todas las típicas características de un IDE más un conjunto de características especiales para el entorno PHP con el objetivo de ayudar a la productividad del desarrollador de PHP. Posee soporte para *Web Services*, PHP4, PHP5 y SQL. Zend Studio permite agilizar el desarrollo Web y permite simplificar proyectos complejos. (INFO&SOFT, 2008)

Algunas de sus características especiales son:

- SCM: Soporte integrado para Subversion.
- DB: Visor de la estructura de base de datos y de contenidos, cliente SQL para diferentes tipos de bases de datos.
- Editor: Coloreo de la sintaxis. Coloreo de errores.
- Editor: Autocompletado de código para funciones PHP y código propio basado en comentarios PHPDoc.
- Editor: Analizador de Código para encontrar problemas típicos en código PHP.
- Editor: Plantillas de código PHP.
- Depurador: Barra del navegador que permite empezar el depurado de errores en cualquier lugar de su aplicación. (Aulke, 2007)

Por todas las bondades mencionadas con anterioridad se escoge el Zend Studio como entorno de desarrollo para llevar a cabo la realización del portal Web, ya que aunque es un software propietario, de los dos IDEs existentes es uno de los que con más experiencia de trabajo se cuenta en la Universidad.

1.5 Servidores Web.

Un servidor Web es una aplicación que sirve archivos de un sitio Web a petición de los usuarios. Se llama así también a la máquina conectada a la red en la que están almacenadas físicamente las páginas que componen un sitio. (HERNÁNDEZ) El servidor Web escucha peticiones HTTP y las satisface, pudiendo ser esta respuesta la búsqueda de una página Web o la ejecución de un programa, devolviendo un resultado al cliente que hizo la petición. En el presente trabajo es esencial la implementación de aplicaciones que se ejecuten en el servidor, ya que se harán modificaciones a los datos de los estudiantes que solo pueden ocurrir en la máquina donde esté montado el sitio, o sea, el Servidor Web.

1.5.1 Internet Information Server.

Internet Information Server (ISS) es una serie de servicios para los ordenadores que funcionan con Windows. Siendo al inicio parte del Option Pack para Windows NT se fue integrando a otros sistemas operativos de Microsoft como Windows 2000 o Windows Server 2003. Los servicios que ofrece son FTP, SMTP, NNTP y HTTP/HTTPS. El Internet Information Server convierte a un ordenador en un servidor de Internet o Intranet pues proporcionan las herramientas y funciones necesarias para administrar de forma sencilla un servidor Web seguro.

1.5.2 Apache.

Es un Servidor Web (HTTP) de código abierto. Actualmente es uno de los servidores más utilizados en todo el mundo y cuenta con versiones para diversas plataformas (GNU/Linux, Windows, OS X...). (FERNÁNDEZ 2006)

Es flexible y eficiente, continuamente actualizado y adaptado a los nuevos protocolos. Entre sus características destacan:

- Multiplataforma.

- Modular: Puede ser adaptado a diferentes entornos y necesidades, con los diferentes módulos de apoyo que proporciona, y con la API de programación de módulos, para el desarrollo de módulos específicos.
- Soporte para manejo de archivos grandes (más grandes de 2 GB) en plataformas de 32 bits.
- Extensible: gracias a ser modular se han desarrollado diversas extensiones entre las que destaca PHP, un lenguaje de programación del lado del servidor.
- Se desarrolla de forma abierta.

1.5.3 ¿Por qué usar Apache?

Se decidió utilizar el servidor Web Apache pues Internet Information Server solo funciona bajo servidores Microsoft, debe ser usado bajo licencia, mientras Apache es un proyecto libre y por tanto gratuito, además es uno de los servidores Web más usado a nivel mundial.

1.6 Metodología Ágil.

El enfoque de metodologías ágiles surgió a principios de la década del '90, y fue bastante revolucionario para su momento, pues iba en contra de todas las creencias que existían y que apuntaban hacia la necesidad insustituible de utilizar procesos altamente definidos. La primera experiencia relevante que dio como resultado la adopción de estas metodologías fue el proyecto "Chrysler Comprehensive Compensation", donde Kent Beck, que luego sería el autor de Extreme Programming, tiró todo el código hasta entonces existente del atrasado proyecto y utilizó técnicas del manifiesto ágil, logrando resultados sorprendentes.

1.6.1 Principios Del manifiesto ágil.

- La prioridad es satisfacer al cliente mediante tempranas y continuas entregas de software que le aporte un valor.
- Dar la bienvenida a los cambios. Se capturan los cambios para que el cliente tenga una ventaja competitiva.

- Entregar frecuentemente software que funcione desde un par de semanas a un par de meses, con el menor intervalo de tiempo posible entre entregas.
- La gente del negocio y los desarrolladores deben trabajar juntos a lo largo del proyecto.
- Construir el proyecto en torno a individuos motivados. Darles el entorno y el apoyo que necesitan y confiar en ellos para conseguir finalizar el trabajo.
- El diálogo cara a cara es el método más eficiente y efectivo para comunicar información dentro de un equipo de desarrollo.
- El software que funciona es la medida principal de progreso.
- La atención continua a la calidad técnica y al buen diseño mejora la agilidad.
- La simplicidad es esencial.
- Las mejores arquitecturas, requisitos y diseños surgen de los equipos organizados por sí mismos.
- En intervalos regulares, el equipo reflexiona respecto a cómo llegar a ser más efectivo, y según esto ajusta su comportamiento.

¿Por qué usar Metodologías Ágiles?

Las metodologías ágiles tienen marcadas diferencias con forma tradicional de desarrollar un software. Estas están basadas en heurísticas provenientes de prácticas de producción de código, están especialmente preparadas para cambios durante el proyecto, basa su equipo en no más de 10 programadores trabajando en el mismo sitio, utiliza menos roles y hace menos énfasis en la arquitectura del software, además de incluir al cliente como un miembro más del equipo, abogando siempre por la comunicación interpersonal y la flexibilidad. El sistema que se desarrolla en el presente trabajo tiene características que imponen utilizar metodologías ágiles, pues la resistencia al cambio, los procesos en extremo controlados, las reuniones protocolares y la cantidad excesiva de roles, entre otras cualidades que poseen las metodologías tradicionales, harían el proceso más lento y más complicado.

Scrum

Scrum es una metodología ágil que define un proceso empírico, iterativo e incremental de desarrollo que intenta obtener ventajas respecto a los procesos definidos (cascada, espiral, prototipos, etc.) mediante la aceptación de la naturaleza caótica del desarrollo de software, y la utilización de prácticas tendientes a manejar el riesgo a niveles aceptables. El mismo surge en 1986, de un artículo de la Harvard Business Review titulado “The New Product Development Game” de Hirotaka Takeuchi e Ikujiro Nonaka, que introducía las mejores prácticas más utilizadas en 10 compañías japonesas altamente innovadoras. A partir de ahí y tomando referencias al juego de rugby, Ken Schwaber y Jeff Sutherland formalizan el proceso conocido como Scrum en el año 1995.

Scrum es un método iterativo e incremental que enfatiza prácticas y valores de project management por sobre las demás disciplinas del desarrollo. Al principio del proyecto se define el Product Backlog, que contiene todos los requerimientos funcionales y no funcionales que deberá satisfacer el sistema a construir. Los mismos estarán especificados de acuerdo a las convenciones de la organización ya sea mediante: features, casos de uso, diagramas de flujo de datos, incidentes, tareas, etc. El Product Backlog será definido durante reuniones de planeamiento con los stakeholders. A partir de ahí se definirán las iteraciones, conocidas como Sprint en la jerga de Scrum, en las que se irá evolucionando la aplicación evolutivamente. Cada Sprint tendrá su propio Sprint Backlog que será un subconjunto del Product Backlog con los requerimientos a ser construidos en el Sprint correspondiente. La duración recomendada del Sprint es de un mes.

Aunque surgió como modelo para el desarrollo de productos tecnológicos, también se emplea en entornos que trabajan con requisitos inestables y que requieren rapidez y flexibilidad; situaciones frecuentes en el desarrollo de determinados sistemas de software.

Crystal Clear

Alistair Cockburn es el propulsor detrás de la serie de metodologías Crystal. Las mismas presentan un enfoque ágil, con gran énfasis en la comunicación, con cierta tolerancia que la hace ideal en los casos en que sea inaplicable la disciplina requerida

por XP. Crystal "Clear" es la encarnación más ágil de la serie y de la que más documentación se dispone. La misma se define con mucho énfasis en la comunicación y de forma muy liviana en relación a los entregables. Crystal maneja iteraciones cortas con feedback frecuente por parte de los usuarios/clientes, minimizando de esta forma la necesidad de productos intermedios. Otra de las cuestiones planteadas es la necesidad de disponer de un usuario real aunque sea de forma *part time* para realizar validaciones sobre la Interface del Usuario y para participar en la definición de los requerimientos funcionales y no funcionales del software.

La familia Crystal dispone un código de color para marcar la complejidad de una metodología: cuanto más oscuro un color, más "pesado" es el método. Cuanto más crítico es un sistema, más rigor se requiere. El código cromático se aplica a una forma tabular elaborada por Cockburn que se usa en muchas metodologías ágiles para situar el rango de complejidad al cual se aplica una metodología.

XP- Extreme Programming.

Es la primera metodología ágil y la que le dio conciencia al movimiento actual de metodologías ágiles. De la mano de Kent Beck, XP ha conformado un extenso grupo de seguidores en todo el mundo, disparando una gran cantidad de libros. Es por tanto, el más destacado de los procesos ágiles de desarrollo de software. Se centra en potenciar las relaciones interpersonales como clave para el éxito en el desarrollo de software, promoviendo el trabajo en equipo, preocupándose por el aprendizaje de los desarrolladores, y propiciando un buen clima de trabajo. XP se basa en realimentación continua entre el cliente y el equipo de desarrollo, comunicación fluida entre todos los participantes, simplicidad en las soluciones implementadas y coraje para enfrentar los cambios. XP se define como especialmente adecuada para proyectos con requisitos imprecisos y muy cambiantes, y donde existe un alto riesgo técnico. A continuación se presentan las características esenciales de XP organizadas en los tres apartados siguientes: historias de usuario, roles, proceso y prácticas.

Las Historias de Usuario.

Son la técnica utilizada para especificar los requisitos del software. Se trata de tarjetas de papel en las cuales el cliente describe brevemente las características que el sistema debe poseer, sean requisitos funcionales o no funcionales. El tratamiento de

las historias de usuario es muy dinámico y flexible. Cada historia de usuario es lo suficientemente comprensible y delimitada para que los programadores puedan implementarla en unas semanas.

Roles XP.

Los roles de acuerdo con la propuesta original de Beck son:

- Programador. El programador escribe las pruebas unitarias y produce el código del sistema.
- Cliente. Escribe las historias de usuario y las pruebas funcionales para validar su implementación. Además, asigna la prioridad a las historias de usuario y decide cuáles se implementan en cada iteración centrándose en aportar mayor valor al negocio.
- Encargado de pruebas (Tester). Ayuda al cliente a escribir las pruebas funcionales. Ejecuta las pruebas regularmente, difunde los resultados en el equipo y es responsable de las herramientas de soporte para pruebas.
- Encargado de seguimiento (Tracker). Proporciona realimentación al equipo. Verifica el grado de acierto entre las estimaciones realizadas y el tiempo real dedicado, para mejorar futuras estimaciones. Realiza el seguimiento del progreso de cada iteración.
- Entrenador (Coach). Es responsable del proceso global. Debe proveer guías al equipo de forma que se apliquen las prácticas XP y se siga el proceso correctamente.
- Consultor. Es un miembro externo del equipo con un conocimiento específico en algún tema necesario para el proyecto, en el que puedan surgir problemas.
- Gestor (Big boss). Es el vínculo entre clientes y programadores, ayuda a que el equipo trabaje efectivamente creando las condiciones adecuadas. Su labor esencial es de coordinación.

Proceso XP.

El ciclo de desarrollo consiste (a grandes rasgos) en los siguientes pasos:

1. El cliente define el valor de negocio a implementar.
2. El programador estima el esfuerzo necesario para su implementación.
3. El cliente selecciona qué construir, de acuerdo con sus prioridades y las restricciones de tiempo.
4. El programador construye ese valor de negocio.
5. Vuelve al paso 1.

En todas las iteraciones de este ciclo tanto el cliente como el programador aprenden. No se debe presionar al programador a realizar más trabajo que el estimado, ya que se perderá calidad en el software o no se cumplirán los plazos. De la misma forma el cliente tiene la obligación de manejar el ámbito de entrega del producto, para asegurarse que el sistema tenga el mayor valor de negocio posible con cada iteración.

El ciclo de vida ideal de XP consiste de seis fases: Exploración, Planificación de la Entrega (Release), Iteraciones, Producción, Mantenimiento y Muerte del Proyecto.

Prácticas XP.

La principal suposición que se realiza en XP es la posibilidad de disminuir la mítica curva exponencial del costo del cambio a lo largo del proyecto, lo suficiente para que el diseño evolutivo funcione. Esto se consigue gracias a las tecnologías disponibles para ayudar en el desarrollo de software y a la aplicación disciplinada de las siguientes prácticas.

- **El juego de la planificación.** Hay una comunicación frecuente el cliente y los programadores. El equipo técnico realiza una estimación del esfuerzo requerido para la implementación de las historias de usuario y los clientes deciden sobre el ámbito y tiempo de las entregas y de cada iteración.
- **Entregas pequeñas.** Producir rápidamente versiones del sistema que sean operativas, aunque no cuenten con toda la funcionalidad del sistema. Esta versión ya constituye un resultado de valor para el negocio. Una entrega no debería tardar más 3 meses.

- **Metáfora.** El sistema es definido mediante una metáfora o un conjunto de metáforas compartidas por el cliente y el equipo de desarrollo. Una metáfora es una historia compartida que describe cómo debería funcionar el sistema (conjunto de nombres que actúen como vocabulario para hablar sobre el dominio del problema, ayudando a la nomenclatura de clases y métodos del sistema).
- **Diseño simple.** Se debe diseñar la solución más simple que pueda funcionar y ser implementada en un momento determinado del proyecto.
- **Pruebas.** La producción de código está dirigida por las pruebas unitarias. Éstas son establecidas por el cliente antes de escribirse el código y son ejecutadas constantemente ante cada modificación del sistema.
- **Refactorización.** Es una actividad constante de reestructuración del código con el objetivo de remover duplicación de código, mejorar su legibilidad, simplificarlo y hacerlo más flexible para facilitar los posteriores cambios. Se mejora la estructura interna del código sin alterar su comportamiento externo.
- **Programación en parejas.** Toda la producción de código debe realizarse con trabajo en parejas de programadores. Esto conlleva ventajas implícitas (menor tasa de errores, mejor diseño, mayor satisfacción de los programadores, etc.).
- **Propiedad colectiva del código.** Cualquier programador puede cambiar cualquier parte del código en cualquier momento.
- **Integración continua.** Cada pieza de código es integrada en el sistema una vez que esté lista. Así, el sistema puede llegar a ser integrado y construido varias veces en un mismo día.
- **40 horas por semana.** Se debe trabajar un máximo de 40 horas por semana. No se trabajan horas extras en dos semanas seguidas. Si esto ocurre, probablemente está ocurriendo un problema que debe corregirse. El trabajo extra desmotiva al equipo.
- **Cliente in-situ.** El cliente tiene que estar presente y disponible todo el tiempo para el equipo. Éste es uno de los principales factores de éxito del proyecto XP. El cliente conduce constantemente el trabajo hacia lo que aportará mayor valor

de negocio y los programadores pueden resolver de manera inmediata cualquier duda asociada. La comunicación oral es más efectiva que la escrita.

- **Estándares de programación.** XP enfatiza que la comunicación de los programadores es a través del código, con lo cual es indispensable que se sigan ciertos estándares de programación para mantener el código legible. (Calderón, 2007)

1.6.2 Selección de la Metodología.

Luego de realizar un estudio detenido de las metodologías ágiles mencionadas se llegó a la conclusión de que la más adaptable al presente trabajo es la metodología XP. El estudio tuvo en cuenta que todas las metodologías estudiadas poseen cualidades destacables como son la sencillez y la adaptabilidad a los cambios. Se decidió utilizar XP porque esta propone que la comunicación y la satisfacción del cliente es lo principal. También potencia el trabajo en grupo y lo hace más dinámico, dando como resultado una entrega a tiempo que cumple con los niveles de calidad requeridos.

1.7 Conclusiones

Con el estudio de las tecnologías que actualmente son utilizadas para el desarrollo de sistemas Web y la importancia de desarrollar sobre diferentes plataformas utilizando las ventajas del código abierto se decidió utilizar metodologías ágiles eligiendo la XP. Se trabajará con el lenguaje de programación PHP con la ayuda de la herramienta Zend Studio for Eclipse, utilizando el sistema gestor de contenido Drupal y el Servidor Web Apache y como gestor de base de datos se utilizará PostgreSQL.

CAPÍTULO 2

Exploración y Planificación

2.1 Introducción

Luego de haber realizado un estudio del arte y de haber seleccionado las herramientas y las metodologías a utilizar, estamos en condiciones de realizar una propuesta de solución de nuestro sistema, basándonos en la metodología ágil XP. Este capítulo está enmarcado en las fases de exploración y planificación definidas en la metodología escogida así como los artefactos que se generan en dichas fases.

2.2 Descripción de las acciones vinculadas al campo de acción

Actualmente en la facultad 8 de la Universidad de las Ciencias Informáticas para los profesores guías se hace muy complicado el proceso de gestionar toda la información relacionada con los estudiantes que atienden, pues la mayoría de esta información se gestiona de forma manual, además esta información no se encuentra centralizada y no existe ningún mecanismo donde se pueda divulgar siendo accesible tanto para profesores como para estudiantes. La facultad 8 cuenta con un coordinador general, que se encarga de la dirección de los profesores guías y al cual se subordinan los coordinadores de año. Cada profesor guía es responsable de llevar la caracterización de los estudiantes de su grupo, trazar el plan de proyecto educativo de la brigada y llevar un control de las actividades que realiza cada estudiante, divididas en actividades docentes, de extensión y políticas. Los estudiantes dan una evaluación del profesor guía que es recogida por el coordinador general. Cada profesor coordinador de año traza una estrategia educativa para el año, por la cual los profesores guías deben guiarse. Los estudiantes coordinadores de año apoyan estas actividades. Toda esta información debe llegar a manos del coordinador general, que es el responsable de orientar y evaluar todo el proceso.

2.2.1 Objetos de automatización

En el proceso de gestionar la información referente a los estudiantes de la facultad 8 de la UCI se realizan numerosas actividades que necesitan ser automatizadas, puesto que se ha hecho muy engorroso el manejo de muchos datos de forma manual, trayendo

consigo pérdida de tiempo y en la mayoría de los casos que se cometan errores innecesarios y que a veces se pierda información. Las tareas que se automatizaran son la preparación de la estrategia educativa de la facultad, del año y de los grupos, que es trazada por la dirección de la facultad, los coordinadores de año y los profesores guías respectivamente. Se permitirá la divulgación de las actividades que se vayan a realizar en la facultad. También con el sistema los estudiantes podrán dar la evaluación al profesor guía. El coordinador general tendrá permisos para subir documentos que necesite estén al alcance de todos y podrá descargar las estrategias educativas de los coordinadores de año y los proyectos educativos de los profesores guías.

2.2.2 Propuesta del sistema

Para la realización del siguiente trabajo se propone la implementación de un sistema que permita la gestión (insertar, modificar y eliminar) de un conjunto de información referente a los estudiantes de la facultad 8. Este sistema será una aplicación Web desarrollada en el CMS Drupal, está constará con una autenticación por roles, brindado servicios en dependencia del tipo de usuario autenticado, logrando la seguridad de los datos que se manejan en el sitio. El sistema brindará la opción de publicar avisos tanto para estudiantes como para profesores, así como un panel de descargas que brindará la opción de subir y descargar archivos de la aplicación fácilmente. También contará con vínculos a otras páginas. Permite agregar las etapas, los años y los grupos docentes que pertenecen a dichas etapa. Además permite la gestión de estudiantes, profesores y coordinadores de año. Se contará con la opción de vincular los estudiantes a un grupo determinado, así como el profesor guía que le corresponde y los coordinadores al año que atiende.

2.3 Fase de Exploración

En esta fase se define el alcance del proyecto y al mismo tiempo el equipo de trabajo se familiariza con las herramientas y tecnologías que se utilizan en el proyecto, realizándose las historias de los usuarios que no son más que la forma de representar los requisitos del sistema a implementar. La duración de esta etapa puede variar en dependencia de la familiarización que tengan los desarrolladores con las herramientas en el caso de la realización del Sitio Web para Profesores Guías de la facultad 8 durara 3 semanas, ya que no es un proyecto de mucha envergadura.

2.3.1 Historias de usuarios

Tabla 2.1 Plantilla de historia de usuario

Historia de Usuario	
No. : Número de la historia de usuario.	Nombre: El nombre de la HU sirve para identificarla fácilmente.
Usuario: es el usuario del sistema que utiliza o protagoniza la historia.	
Prioridad en Negocio (Alta / Media / Baja): Cuán importante es para el cliente.	Puntos Estimados: Tiempo de duración relacionada con la métrica de estimación.
Riesgos en Desarrollo (Alta / Media / Baja): Cuán difícil es para el desarrollador.	Iteración Asignada: La iteración (liberación en nuestro proceso) a la que corresponde.
Descripción: La descripción de la historia de usuario, detallando las operaciones del usuario y opcionalmente la respuesta del sistema.	
Observaciones: Información de interés, como glosarios o información de los usuarios.	

Tabla 2.2 HU Gestionar cuenta de usuario.

Historia de Usuario	
No. :1	Nombre: Gestionar cuenta de usuario.
Usuario: Administrador	
Prioridad en Negocio (Alta / Media / Baja): Alta	Puntos Estimados:
Riesgos en Desarrollo (Alta / Media / Baja): Media	Iteración Asignada: 1.
Descripción: El administrador realiza las acciones de crear, modificar y eliminar las cuentas y perfiles del usuario (profesor guía, estudiantes, coordinadores de año) en el sistema. Al entrar al sistema al administrador se le brinda la posibilidad de crear cuentas para la posterior asignación de permisos dentro del sistema. Se brinda la posibilidad de editar los perfiles después de llenados los datos (nombre, CI, provincia, municipio, grupo, momento, nivel, correo) e igualmente eliminar la cuenta.	

Observaciones:

Tabla 2.3 HU Autenticar.

Historia de Usuario	
No. : 3	Nombre: Autenticar.
Usuario: Todos.	
Prioridad en Negocio (Alta / Media / Baja): Alta.	Puntos Estimados:
Riesgos en Desarrollo (Alta / Media / Baja): Media.	Iteración Asignada: 1.
Descripción: Se brinda la opción a los usuarios de autenticarse comprobando su usuario y contraseña brindándole los servicios acorde a sus roles específicos dentro de la aplicación (Estudiante, Profesor, Profesor guía, Coordinador de año, Dirección).	
Observaciones:	

Tabla 2.4 HU Gestionar proyecto educativo.

Historia de Usuario	
No. : 4	Nombre: Gestionar proyecto educativo.
Usuario: Dirección.	
Prioridad en Negocio (Alta / Media / Baja): Alta.	Puntos Estimados:
Riesgos en Desarrollo (Alta / Media / Baja): Media	Iteración Asignada: 1
Descripción: Se brinda la posibilidad de subir al sitio la planilla Estrategia Educativa que traza la facultad, así como el proyecto educativo del año y el proyecto educativo de la brigada, trazado por el coordinador de año y el profesor guía respectivamente.	
Observaciones:	

Tabla 2.5 HU Consultar proyecto educativo.

Historia de Usuario	
No. : 5	Nombre: Consultar proyecto educativo.
Usuario: todos	

Prioridad en Negocio (Alta / Media / Baja): Alta	Puntos Estimados:
Riesgos en Desarrollo (Alta / Media / Baja): Media	Iteración Asignada: 1
Descripción: Se brinda la posibilidad de que todos los usuarios puedan consultar y descargar la estrategia educativa de la facultad, así como el proyecto educativo del año y el proyecto educativo de la brigada.	
Observaciones:	

Tabla 2.6 HU Gestionar Grupos

Historia de Usuario	
No. : 6	Nombre: Gestionar Grupos
Usuario: Administrador.	
Prioridad en Negocio (Alta / Media / Baja): Alta.	Puntos Estimados:
Riesgos en Desarrollo (Alta / Media / Baja): Media.	Iteración Asignada: 1
Descripción: Se debe brindar la posibilidad al usuario Dirección de crear, eliminar y modificar grupos.	
Observaciones:	

Tabla 2.7 HU Gestionar descargas.

Historia de Usuario	
No. : 8	Nombre: Gestionar descargas
Usuario: Dirección	
Prioridad en Negocio (Alta / Media / Baja): Alta.	Puntos Estimados:
Riesgos en Desarrollo (Alta / Media / Baja): Media.	Iteración Asignada: 1
Descripción: Se permitirá subir archivos al sistema.	
Observaciones: Todos los usuarios podrán descargar estos archivos, los cuales contendrán todo tipo de información de interés que el administrador quiera poner a	

disposición de ellos.

Tabla 2.8 HU Gestionar años.

Historia de Usuario	
No. :9	Nombre: Gestionar años
Usuario: Dirección	
Prioridad en Negocio (Alta / Media / Baja): Alta.	Puntos Estimados:
Riesgos en Desarrollo (Alta / Media / Baja): Baja.	Iteración Asignada: 1
Descripción: Brinda la posibilidad a la dirección de gestionar los años que estarán presentes en la facultad.	
Observaciones: La entrada de años será flexible para el caso de que la facultad cambie su política y adicione o quite años, por ejemplo que trabaje solo con cuarto y quinto año.	

Tabla 2.9 HU Gestionar curso.

Historia de Usuario	
No. :11	Nombre: Gestionar curso
Usuario: Dirección.	
Prioridad en Negocio (Alta / Media / Baja): Alta.	Puntos Estimados:
Riesgos en Desarrollo (Alta / Media / Baja): Media.	Iteración Asignada: 1
Descripción: Se brinda la posibilidad al usuario con el rol de Dirección de consultar, agregar, modificar y eliminar un curso. Ejemplo 2009-2010.	
Observaciones:	

Tabla 2.10 HU Gestionar departamento.

Historia de Usuario	
No. :13	Nombre: Gestionar departamento
Usuario: Dirección.	

Prioridad en Negocio (Alta / Media / Baja): Alta.	Puntos Estimados:
Riesgos en Desarrollo (Alta / Media / Baja): Media.	Iteración Asignada: 1
Descripción: Se brindará la posibilidad de agregar los diferentes departamentos a los que pertenecen los profesores de la facultad, ejemplo: Departamento de Humanidades, Programación, etc.	
Observaciones:	

Tabla 2.11 HU Consultar departamento.

Historia de Usuario	
No. : 14	Nombre: Consultar departamento
Usuario: Dirección.	
Prioridad en Negocio (Alta / Media / Baja): Alta.	Puntos Estimados:
Riesgos en Desarrollo (Alta / Media / Baja): Media.	Iteración Asignada: 1
Descripción: Se permitirá la opción de que tanto estudiantes como profesores puedan ver los diferentes departamentos que existen en la facultad y los profesores que lo integran.	
Observaciones:	

Tabla 2.12 HU Gestionar procedencia.

Historia de Usuario	
No. : 15	Nombre: Gestionar procedencia
Usuario: Dirección.	
Prioridad en Negocio (Alta / Media / Baja): Alta.	Puntos Estimados:
Riesgos en Desarrollo (Alta / Media / Baja): Media.	Iteración Asignada: 1
Descripción: Se brindará la opción de poder agregarle a los estudiantes la procedencia, que no es más que el lugar de donde ellos ingresaron a la escuela, por	

ejemplo: IPUEC, IPI, IPVCE, entre otros.
Observaciones:

Tabla 2.13 HU Consultar procedencia.

Historia de Usuario	
No. : 16	Nombre: Consultar procedencia
Usuario: Dirección.	
Prioridad en Negocio (Alta / Media / Baja): Alta.	Puntos Estimados:
Riesgos en Desarrollo (Alta / Media / Baja): Media.	Iteración Asignada: 1
Descripción: Se les permitirá a todos los usuarios ver cual es la procedencia de los estudiantes.	
Observaciones:	

Para continuar el análisis de las mismas remitirse a los anexos.

2.4 Fase de Planificación

En XP se plantea que la planificación debe ser un proceso de constante intercambio entre el cliente y los desarrolladores del proyecto. En esta fase el cliente establece la prioridad de cada historia de usuario, y posteriormente, los desarrolladores son los encargados de realizar una estimación del esfuerzo necesario de cada una de ellas. Luego se toman acuerdos sobre el contenido de la primera entrega y se determina un cronograma en conjunto con el cliente.

Las estimaciones de esfuerzo asociado a la implementación de las historias la establecen los programadores utilizando como métrica medida el punto. Un punto se considera como una semana ideal de trabajo, donde los miembros de los equipos de desarrollo trabajan el tiempo planeado sin ningún tipo de interrupción. Las historias generalmente valen de uno a tres puntos. Los desarrolladores deberán hacer una estimación de cuánto tiempo, idealmente, les llevará implementar cada historia de usuario. Las condiciones ideales son aquellas en las que se codifica la historia de usuario sin otras distracciones y sabiendo exactamente qué es lo que hay que implementar.

2.4.1 Estimación de esfuerzos por historias de usuario

Tabla 2.29 Estimación de esfuerzos por Historia de Usuario

Historia de usuario	Puntos de estimación
Gestionar cuenta de usuario	2
Consultar perfil de usuario	1
Autenticar	1
Consultar cursos	1
Gestionar cursos	2
Consultar avisos	1
Gestionar avisos	2
Consultar noticias	1
Gestionar noticias	2
Gestionar descargas	1
Consultar vínculos	1
Gestionar vínculos	2
Consultar años	1
Gestionar años	1
Consultar departamentos	1
Gestionar departamentos	2
Consultar procedencia	1
Gestionar procedencia	2
Consultar proyectos educativos	1
Gestionar proyectos educativos	1
Gestionar persona	2
Consultar persona	3
Consultar grupos	1
Gestionar grupos	1

Vincular grupo a un curso en un semestre	3
Vincular profesor guía al grupo	3
Vincular estudiantes al grupo	3

2.4.2 Plan de iteraciones

Después de haber identificado las historias de usuarios y haber realizado una previa planificación y estimación de esfuerzos por historias de usuarios se procede a la planificación de la etapa de implementación del sistema. En este plan se propone la prioridad con que se irán implementando las historias de usuarios organizadas por iteraciones así como las posibles fechas de liberación. Los elementos que deben tomarse en cuenta durante la elaboración del Plan de la Iteración son: historias de usuario no abordadas, velocidad del proyecto, pruebas de aceptación no superadas en la iteración anterior y tareas no terminadas en la iteración anterior. Al principio de cada iteración se realizan las tareas de análisis con el cliente, recaudando todos los datos posibles, por lo que el cliente debe participar de forma activa en esta fase del ciclo.

2.4.3 Plan de duración de las iteraciones

1ra Iteración.

Esta iteración tendrá como objetivo darle cumplimiento a las historias de usuarios que tengan prioridad alta, estas historias de usuario son 1, 3, 4, 5, 6, 8, 9, 11, 13, 14, 15, 16, que son las de mayor importancia para el cliente, estas recogen las principales funcionalidades del sistema, las cuales hacen alusión de manera general a la administración de cuentas y perfiles de usuario, la autenticación de los usuarios permitiendo la seguridad del sitio, la gestión y consulta de la estrategia educativa de la facultad, del proyecto educativo del año y del proyecto educativo de la brigada, así como la gestión y consulta de las etapas, gestión y consulta de grupos y años. Además se le brindará al cliente una versión de prueba, con el objetivo de obtener una retroalimentación beneficiosa para el grupo de desarrollo.

2da Iteración.

En esta iteración se le dará cumplimiento a las historias de usuario 2, 7, 10, 12, 17, 19, 20, 21, estas son las historias de usuario con prioridad media en el negocio. En estas se

le da cumplimiento a las peticiones del cliente descritas en estas historias de usuario como son consultar perfil de usuario, consultar grupos, consultar años, consultar etapas, gestionar persona que no es más que gestionar los diferentes usuarios que tendrá el sistema, vincular grupos a una etapa determinada, vincular el profesor guía al grupo que le corresponde, vincular estudiantes a un grupo. En esta etapa se le presentará al cliente la versión de prueba con el objetivo de obtener su opinión y cambios en caso de que exista la necesidad.

3ra Iteración.

En esta última iteración se agrupan las historias de usuario con prioridad baja del negocio, entre ellas se encuentran las HU 18, 22, 23, 24, 25, 26, 27, en estas se le da cumplimiento a las peticiones del cliente como son consultar persona, gestionar y consultar vínculos, gestionar y consultar avisos y noticias publicadas en el sitio. Con la culminación de esta iteración se tendrá la última versión del producto, poniéndose a prueba por un periodo de tiempo para evaluar el desempeño del sitio.

Plan de duración de las iteraciones.

En este plan se especifica más detalladamente el orden de desarrollo de las historias de usuarios dentro de cada iteración así como la estimación completa de dicha iteración.

Tabla 2.30 Plan de duración de las iteraciones.

Iteraciones	Orden de las historias de usuario a implementar	Duración de las iteraciones.
1ra Iteración	<ol style="list-style-type: none"> 1. Gestionar cuenta de usuario. 2. Autenticar. 3. Gestionar proyecto educativo. 4. Consultar proyecto educativo. 5. Gestionar Grupos. 6. Gestionar descargas. 7. Gestionar años. 8. Gestionar curso. 9. Gestionar departamento. 10. Consultar departamento. 	4 semanas

	<ul style="list-style-type: none"> 11. Gestionar procedencia. 12. Consultar procedencia. 	
2da Iteración	<ul style="list-style-type: none"> 1. Consultar perfil de usuario. 2. Consultar grupos. 3. Consultar años. 4. Gestionar persona. 5. Consultar curso. 6. Vincular grupo a un curso en un semestre. 7. Vincular profesor guía a un grupo. 8. Vincular estudiantes al grupo. 	2 semanas
3ra Iteración	<ul style="list-style-type: none"> 1. Gestionar vínculos. 2. Consultar vínculos. 3. Consultar persona. 4. Gestionar avisos. 5. Consultar avisos. 6. Gestionar noticias. 7. Consultar noticias. 	3 semanas

2.4.4 Plan de entregas

En este plan se realiza el cronograma de entregas que establece qué historias de usuario serán agrupadas para conformar una entrega, y el orden de las mismas. En este plan se acoplan las funcionalidades referentes a un mismo tema en módulos, esto permite un mayor entendimiento en la fase de implementación quedando de la siguiente manera:

Tabla 2.31 Módulos y Historias de Usuario abarcadas

Módulos	Historia de usuario
Curso	Consultar cursos
	Gestionar cursos

Aviso	Consultar avisos
	Gestionar avisos
Noticia	Consultar noticias
	Gestionar noticias
Descarga	Gestionar descargas
Vinculo	Consultar vínculos
	Gestionar vínculos
Año	Consultar años
	Gestionar años
Departamento	Consultar departamentos
	Gestionar departamentos
Procedencia	Consultar procedencia
	Gestionar procedencia
Proyecto Educativo	Consultar proyectos educativos
	Gestionar proyectos educativos
Persona	Consultar persona
	Gestionar persona
	Autenticar
	Gestionar cuenta de usuario
	Consultar perfil de usuario
Grupo	Consultar grupos
	Gestionar grupos
	Vincular grupo a un curso en un semestre
	Vincular profesor guía al grupo
	Vincular estudiantes al grupo

Se da paso al plan de duración de entregas en el cual se especifican un aproximado de las fechas para cada iteración.

Tabla 2.32 Plan de duración de entrega.

Módulos	1ra Iteración. Mes de marzo	2da Iteración. 1ra y 2da semana de abril	3ra Iteración. 3ra semana de abril y 1ra de mayo
Etapa	V1.0	Terminado	Terminado
Noticia	V1.0	Terminado	Terminado
Avisos	V1.0	Terminado	Terminado
Descarga	V1.0	Terminado	Terminado
Vinculo	V1.0	Terminado	Terminado
Año	V1.0	Terminado	Terminado
Persona	V1.0	V1.1	Terminado
Procedencia	V1.0		
Departamento	V1.0		
Grupo	V1.0	V1.1	Terminado

2.5 Conclusiones

Durante el desarrollo de este capítulo se expusieron los artefactos generados en las fases de exploración y planificación de la metodología XP. Además se especificaron los usuarios que están relacionados con el sistema. Se realizó un levantamiento de las historias de usuarios, donde el cliente precisó la prioridad de cada una de estas historias, definiendo así el orden en el que serán implementadas. El portal cuenta con un total de veintitrés historias de usuarios las cuales serán implementadas en tres iteraciones. En el presente capítulo se hizo un prototipo de interfaz del portal que se va a implementar.

CAPÍTULO 3

Implementación y Prueba

3.1 Introducción

Según la Metodología XP, la implementación debe realizarse de forma iterativa e incremental, permitiendo así que al final de cada iteración surja un producto funcional que debe ser probado y mostrado al cliente, permitiendo de esta forma lograr una constante retroalimentación desarrolladores-cliente, este producto será un producto previamente probado para incrementar la visión de los desarrolladores y el cliente de posibles cambios y soluciones. En el presente capítulo se detallan las tres iteraciones llevadas a cabo durante la etapa de construcción del sistema, exponiéndose las tareas generadas por cada historia de usuario, así como las pruebas de aceptación realizadas al sistema.

3.2 Diseño del sistema

La metodología XP para el diseño de las aplicaciones no requiere la presentación del sistema mediante diagramas de clases utilizando notación UML, en su lugar se usan otras técnicas como las tarjetas CRC (Contenido, Responsabilidad y Colaboración). No obstante el uso de estos diagramas puede aplicarse siempre y cuando influyan en el mejoramiento de la comunicación, no sea un peso su mantenimiento, no sean extensos y se enfoquen en la información importante.

3.2.1 Módulos de Drupal

La forma de diseño y organización que se adopta es de diseñar una tarjeta CRC (Clase-Responsabilidad-Colaboración) por cada uno de los módulos que brindan una funcionalidad directa al negocio, es decir aquellos que fueron desarrollados desde la raíz. De esta forma se obtiene un diseño simple y no se implementan características que no son necesarias. Estas tarjetas CRC permiten desprenderse del método de trabajo basado en procedimientos y trabajar con una metodología basada en objetos.

Tabla 3.1 Tarjeta CRC Módulo Curso

Módulo Curso	
Funcionalidades	Colaboraciones
Insertar curso Modificar curso Eliminar curso	

Tabla 3.2 Tarjeta CRC Módulo Aviso

Módulo Aviso	
Funcionalidades	Colaboraciones
Insertar aviso Modificar aviso Eliminar aviso	

Tabla 3.3 Tarjeta CRC Módulo Noticias

Módulo Noticias	
Funcionalidades	Colaboraciones
Insertar noticia Eliminar noticia Modificar noticia	

Tabla 3.4 Tarjeta CRC Módulo Descargas

Módulo Descarga	
Funcionalidades	Colaboraciones
Insertar descarga Modificar descarga Eliminar descarga	

Tabla 3.5 Tarjeta CRC Módulo Vínculos

Módulo Vínculo	
Funcionalidades	Colaboraciones
Insertar vinculo Modificar vinculo Eliminar vinculo	

Tabla 3.6 Tarjeta CRC Módulo Año

Módulo Año	
Funcionalidades	Colaboraciones
Insertar año Modificar año Eliminar año	

Tabla 3.7 Tarjeta CRC Módulo Departamento

Módulo Departamento	
Funcionalidades	Colaboraciones
Insertar departamento Modificar departamento Eliminar departamento	

Tabla 3.8 Tarjeta CRC Módulo Procedencia

Módulo Procedencia	
Funcionalidades	Colaboraciones
Insertar procedencia Modificar procedencia Eliminar procedencia	

Tabla 3.9 Tarjeta CRC Módulo Proyecto Educativo

Módulo Proyecto Educativo	
Funcionalidades	Colaboraciones
Insertar proyecto educativo	Grupo
Modificar proyecto educativo	Curso
Eliminar proyecto educativo	Año

Tabla 3.10 Tarjeta CRC Módulo Persona

Módulo Persona	
Funcionalidades	Colaboraciones
Adicionar perfil de usuario	Departamento
Editar perfil de usuario	Procedencia
Eliminar perfil de usuario	Grupo
Autenticarse	

Tabla 3.11 Tarjeta CRC Módulo Grupo

Módulo Grupo	
Funcionalidades	Colaboraciones
Insertar grupo	Curso
Modificar grupo	
Eliminar grupo	
Vincular grupo a un curso en un semestre	
Vincular un profesor guía a un grupo	
Vincular los estudiantes a los grupos	

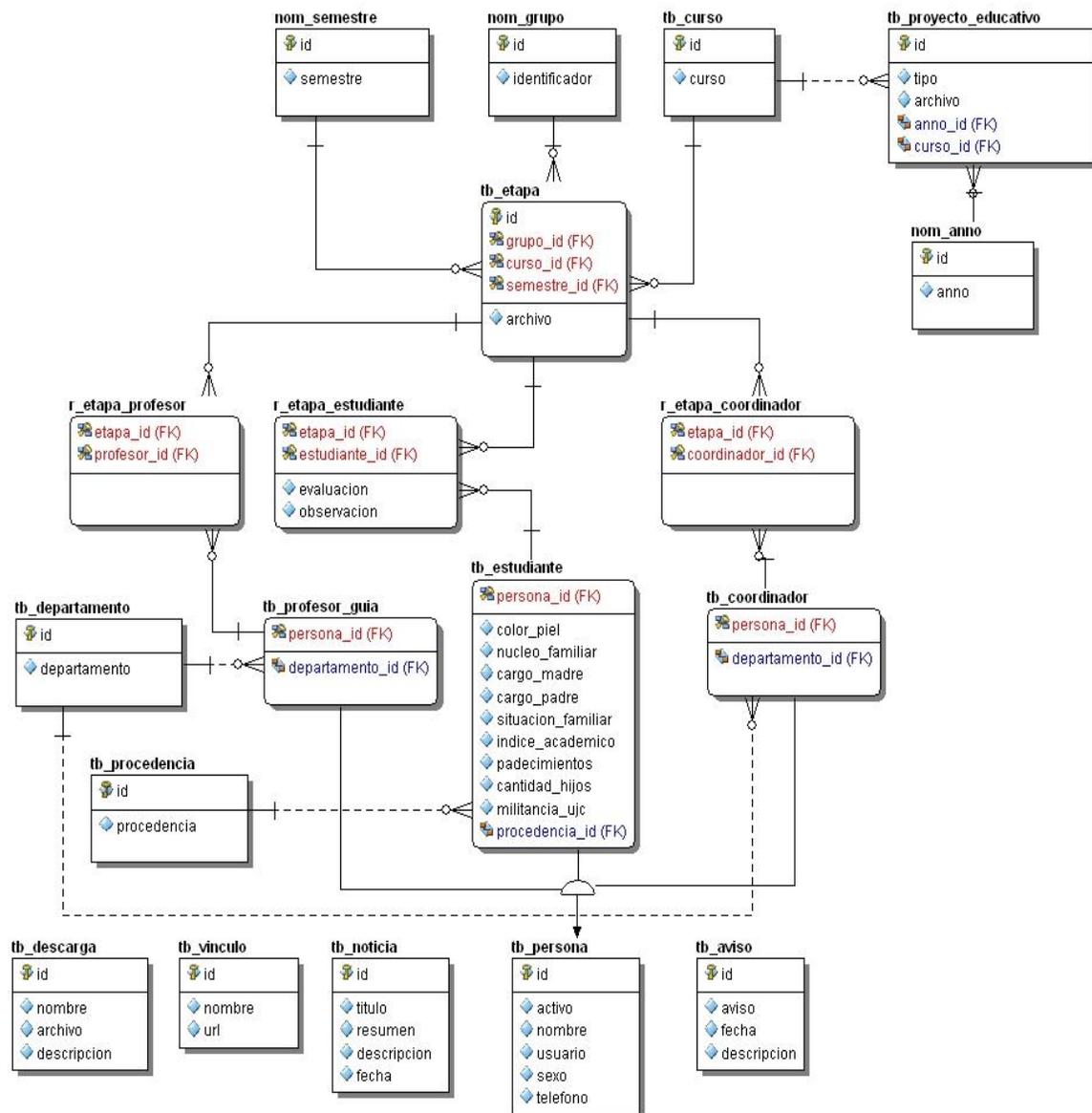
3.3 Diseño de la Base de Datos

El diseño de la base de datos es uno de los diagramas de UML que no se debe pasar por alto, ya que es muy importante dentro del desarrollo de cualquier sistema de

gestión. Dentro de sus principales objetivos está representar la persistencia del negocio expuesto en el capítulo anterior.

El modelo de datos en cuestión posee un nivel de complejidad alto, como consecuencia de que la cantidad de tablas existentes es numerosa así como las relaciones entre ellas. A continuación se muestra el modelo de datos que se utilizó:

Figura 1 Modelo de Datos



3.4 Fase de Implementación.

En esta fase XP plantea la implementación de cada una de las historias de usuarios. Es por esto que al principio de éstas, se lleva a cabo una revisión del plan de iteraciones y se modifica en caso de ser necesario. Como parte de este plan se crean tareas para ayudar a organizar la implementación exitosa de las HU. Estas tareas pueden escribirse en un lenguaje técnico, al contrario de las historias de usuario que son escritas en el lenguaje del cliente. Tomando como referencia la planificación realizada anteriormente, se llevaron a cabo tres iteraciones de desarrollo sobre el sistema, obteniéndose como finalidad un producto con todas las restricciones y características deseadas para ser utilizado. A continuación se muestran cada una de estas iteraciones.

3.4.1 1ra Iteración

Esta iteración tendrá como objetivo darle cumplimiento a las historias de usuarios que serán de vital importancia para el sitio, pues con las mismas se conforma la estructura del negocio ya que son las historias de mayor interés para el cliente. Éstas son las que recogen las principales funcionalidades del sistema.

Tabla 3.12 Módulos tratados en la primera iteración

Módulo	Historias de usuario	Tiempo de Implementación	
		Estimación	Real
Persona	Gestionar cuenta de usuario.	2	1
	Autenticar.	1	0.5
Procedencia	Gestionar procedencia.	2	1
	Consultar procedencia	1	0.5
Grupo	Gestionar grupos.	1	0.5
Proyecto Educativo	Gestionar proyecto educativo.	1	0.5
	Consultar proyecto educativo.	1	0.6
Año	Gestionar año.	1	0.5
Descarga	Gestionar descarga.	1	0.3
Curso	Gestionar curso.	1	0.5

Departamento	Gestionar departamento.	2	1
	Consultar departamento.	1	1

A continuación se muestran las tareas realizadas por cada módulo de esta iteración.

Módulo Persona

Tabla 3.13 Tarea 1 del módulo Persona.

Tarea	
Número de tarea: 1	Número de HU: 1
Nombre de la Tarea: Configuración del módulo usuario	
Tipo de Tarea: Configuración-Desarrollo	Puntos de estimación: 0.5
Fecha de inicio: 20 de Febrero del 2010	Fecha fin: 22 de Febrero del 2010
Programador responsable: Danais Rojas Suárez	
Descripción: Se especificarán los datos que contendrá el perfil de usuario (profesor guía, estudiantes, coordinadores de año) junto con sus cuentas para acceder a la aplicación. Se crearán los roles. Todos estos datos serán almacenados en la base de datos.	

Tabla 3.14 Tarea 2 del módulo Persona.

Tarea	
Número de tarea: 2	Número de HU: 1
Nombre de la Tarea: Configuración del módulo usuario	
Tipo de Tarea: Desarrollo	Puntos de estimación: 0.5
Fecha de inicio: 23 de Febrero del 2010	Fecha fin: 26 de Febrero del 2010
Programador responsable: Danais Rojas Suárez	
Descripción: Se gestionarán los datos contenidos en el perfil de usuario. De esta forma la aplicación posibilita la agregación, modificación y eliminación de cada usuario dentro de la BD. Además permitirá aplicar los distintos tipos de roles a los usuarios, de manera que un rol se le puede aplicar a varios usuarios a la vez.	

Tabla 3.15 Tarea 3 del módulo Persona.

Tarea	
Número de tarea: 3	Número de HU: 3
Nombre de la Tarea: Autenticación	
Tipo de Tarea: Configuración-Desarrollo	Puntos de estimación: 0.5
Fecha de inicio: 27 de Febrero del 2010	Fecha fin: 2 de Marzo del 2010
Programador responsable: Danais Rojas Suárez	
Descripción: Se especificará el modo de autenticación del usuario validando que el usuario y contraseña sean válidos; si son válidos, la aplicación lo agregará automáticamente como un usuario dentro del Drupal.	

Para darle seguimiento a las tareas realizadas para ayudar a la implementación de los módulos de la primera iteración remítase a los anexos.

3.4.2 2da Iteración

En esta iteración se culminará la implementación de los módulos Persona, Curso, Año, Grupo ya que sus funcionalidades fueron divididas en las dos primeras iteraciones, para evitar la sobrecargar de trabajo en la 1ra iteración. Estas historias de usuario son las de prioridad media.

Tabla 3.29 Módulos tratados en la segunda iteración

Módulo	Historias de usuario	Tiempo de Implementación	
		Estimación	Real
Grupo	Consultar grupos	1	0.2
	Vincular grupo a un curso en un semestre	3	2
	Vincular profesor guía a un grupo	3	2
	Vincular estudiantes al grupo	3	2

Persona	Consultar perfil de usuario	1	0.5
	Gestionar persona	2	1
Curso	Consultar curso	1	0.3
Año	Consultar año	1	0.3

Módulo Grupo

Tabla 3.30 Tarea 2 del módulo Grupo.

Tarea	
Número de tarea: 2	Número de HU: 7
Nombre de la Tarea: Configuración de los permisos para el módulo grupo.	
Tipo de Tarea: Desarrollo	Puntos de estimación: 0.5
Fecha de inicio: 4 de Abril del 2010	Fecha fin: 5 de Abril del 2010
Programador responsable: Danais Rojas Suarez	
Descripción: Se asignaran los permisos correspondientes a los roles que tendrán acceso a este módulo.	

Tabla 3.31 Tarea 3 del módulo Grupo.

Tarea	
Número de tarea: 3	Número de HU: 19
Nombre de la Tarea: Vincular un grupo a un curso en un semestre.	
Tipo de Tarea: Desarrollo	Puntos de estimación: 0.5
Fecha de inicio: 6 de Abril del 2010	Fecha fin: 11 de Abril del 2010
Programador responsable: Danais Rojas Suarez	
Descripción: Se implementará la funcionalidad que permitirá poder vincular un grupo a un curso determinado en un semestre dado.	

Tabla 3.32 Tarea 4 del módulo Grupo.

Tarea	
Número de tarea: 4	Número de HU: 20
Nombre de la Tarea: Vincular un profesor guía a un grupo determinado.	

Tipo de Tarea: Desarrollo	Puntos de estimación: 0.5
Fecha de inicio: 12 de Abril del 2010	Fecha fin: 14 de Abril del 2010
Programador responsable: Danais Rojas Suarez	
Descripción: Se implementará la funcionalidad que le permitirá al administrador vincular un profesor ya registrado a un grupo.	

Tabla 3.33 Tarea 5 del módulo Grupo.

Tarea	
Número de tarea: 5	Número de HU: 21
Nombre de la Tarea: Vincular estudiantes a un grupo.	
Tipo de Tarea: Desarrollo	Puntos de estimación: 0.5
Fecha de inicio: 15 de Abril del 2010	Fecha fin: 17 de Abril del 2010
Programador responsable: Danais Rojas Suarez	
Descripción: Se implementará la funcionalidad que le permitirá al administrador vincular los estudiantes a un grupo determinado.	

Para darle seguimiento a las tareas realizadas para ayudar a la implementación de los módulos de la segunda iteración remítase a los anexos.

3.4.3 3ra Iteración

En esta iteración se implementan las historias de usuario con baja prioridad. Estas funcionalidades reflejan en la aplicación un ambiente más amigable y cómodo para el cliente.

Tabla 3.38 Módulos tratados en la tercera iteración

Módulo	Historias de usuario	Tiempo de Implementación	
		Estimación	Real
Vínculos	Gestionar vínculos	2	1
	Consultar vínculos	1	0.5

Noticias	Gestionar noticias	2	1
	Consultar noticias	1	0.3
Avisos	Gestionar avisos	2	1.2
	Consultar avisos	1	0.4
Persona	Consultar persona	3	2

Módulo Vínculos

Tabla 3.39 Tarea 1 del módulo Vínculos.

Tarea	
Número de tarea: 1	Número de HU: 22
Nombre de la Tarea: Gestionar vínculos.	
Tipo de Tarea: Desarrollo	Puntos de estimación: 1
Fecha de inicio: 26 de Abril del 2010	Fecha fin: 27 de Abril del 2010
Programador responsable: Danais Rojas Suárez	
Descripción: Se implementará el módulo vínculos.	

Tabla 3.40 Tarea 2 del módulo Vínculos.

Tarea	
Número de tarea: 2	Número de HU: 22
Nombre de la Tarea: Configuración del módulo vínculos	
Tipo de Tarea: Configuración	Puntos de estimación: 1
Fecha de inicio: 27 de Abril del 2010	Fecha fin: 28 de Abril del 2010
Programador responsable: Danais Rojas Suárez	
Descripción: Se activarán permisos para el acceso al módulo vínculos y a los diferentes vínculos que serán publicados.	

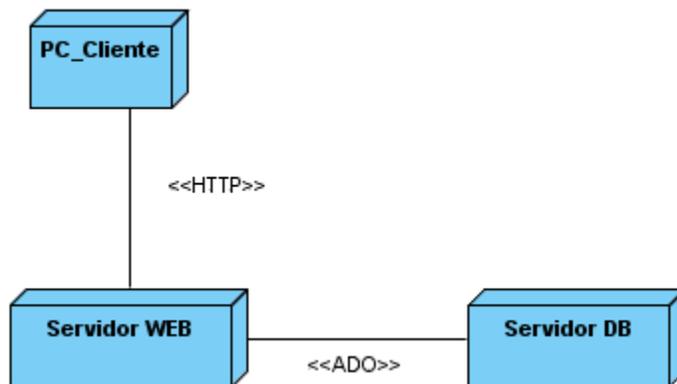
Para darle seguimiento a las tareas realizadas para ayudar a la implementación de los módulos de la tercera iteración remítase a los anexos.

3.4.4 Diagrama de Despliegue

El diagrama de despliegue representa las relaciones físicas entre componentes de software y hardware en el sistema, en otras palabras permite apreciar de forma visual cómo se encuentran relacionados físicamente los componentes de la aplicación. Al igual que el diagrama de componente XP no plantea el modelado de este artefacto pero para un mejor entendimiento del problema propuesto se decide su realización.

En este caso la aplicación se encuentra hospedada en un servidor Web y la misma se comunica con un sistema de gestión de base de datos PostgreSQL.

Figura 2 Diagrama de Despliegue



3.5 Pruebas

Uno de los pilares de la metodología XP es el proceso de pruebas. Esta anima a los programadores a probar constantemente tanto como sea posible. Esto permite aumentar la calidad de los sistemas reduciendo el número de errores no detectados y disminuyendo el tiempo transcurrido entre la aparición de un error y su detección. También permite aumentar la seguridad de evitar efectos colaterales no deseados a la hora de realizar modificaciones y refactorizaciones.

XP divide las pruebas del sistema en dos grupos: pruebas unitarias, encargadas de verificar el código y diseñada por los programadores, y pruebas de aceptación o pruebas funcionales destinadas a evaluar si al final de una iteración se consiguió la funcionalidad requerida diseñadas por el cliente final.

Las pruebas del sistema tienen como objetivo verificar la funcionalidad del sistema a través de sus interfaces externas comprobando que dicha funcionalidad sea la esperada en función de los requisitos del sistema. Generalmente las pruebas del sistema son desarrolladas por los programadores para verificar que su sistema se comporta de la manera esperada, por lo que podrían encajar dentro de la definición de pruebas unitarias que propone XP. (RODRIGUEZ, 2007)

3.5.1 Pruebas de aceptación

Las pruebas de aceptación son creadas a partir de las historias de usuario. Durante una iteración la historia de usuario seleccionada se convertirá en una prueba de aceptación. El cliente especifica los aspectos a testear cuando una historia de usuario ha sido correctamente implementada.

Una historia de usuario puede tener más de una prueba de aceptación, tantas como sean necesarias para garantizar su correcto funcionamiento y no se considera completa hasta que no supera sus pruebas de aceptación. Una prueba de aceptación no es más que una prueba de caja negra. Cada una de ellas representa una salida esperada del sistema. Es responsabilidad del cliente verificar la corrección de las pruebas de aceptación y tomar decisiones a cerca de las mismas. (RODRIGUEZ, 2007)

Tabla 3.46 Prueba1 al módulo persona.

Caso de prueba de aceptación	
Código: HU3_P1	Historia de usuario: 3
Nombre: Autenticar	
Descripción: La aplicación debe brindar la opción de autenticar por dominio uci. Saber identificar a que rol pertenece dicho usuario y en dependencia de su rol los privilegios adecuados para su entrada a la aplicación.	
Condiciones de ejecución : Debe insertar un usuario del dominio uci valido. Debe insertar la contraseña de dicho usuario correctamente.	
Entrada/ Pasos de ejecución: Se intenta entrar en la aplicación.	
Resultado esperado: Que la autenticación se realice correctamente y la asignación de permisos sea acorde la rol que le corresponde.	

Evaluación de la prueba: Prueba satisfactoria.

Tabla 3.47 Prueba 2 al módulo persona.

Caso de prueba de aceptación	
Código: HU17_P1	Historia de usuario: 17
Nombre: Gestionar persona	
Descripción: Prueba para insertar, modificar y eliminar una persona de la aplicación.	
Condiciones de ejecución: El usuario debe estar autenticado y su usuario debe ser Dirección.	
Entrada/ Pasos de ejecución: Se intenta gestionar el perfil de una persona (profesor guía, estudiante, profesor, profesor coordinador de año).	
Resultado esperado: Que se realice la opción de agregar, modificar y eliminar satisfactoriamente una persona.	
Evaluación de la prueba: Prueba satisfactoria	

Tabla 3.48 Prueba 3 al módulo persona.

Caso de prueba de aceptación	
Código: HU18_P1	Historia de usuario: 18
Nombre: Consultar persona	
Descripción: Prueba para listar los usuarios existentes y ver sus datos.	
Condiciones de ejecución: Los datos de los usuarios serán reales.	
Entrada/ Pasos de ejecución: Se intenta listar los usuarios para poder consultar sus datos y poder ordenarlos por diversos criterios.	
Resultado esperado: Que se listes todos los usuarios registrados y se puedan organizar.	
Evaluación de la prueba: Prueba satisfactoria	

3.6 Conclusiones

En este capítulo se realizó el modelo necesario para desarrollar el proceso de implementación del sistema. Se diseñaron todas las clases persistentes que permitieron realizar el diagrama de entidad-relación en el sistema de gestión de bases de datos que se utilizará. Además se desarrollaron las tareas correspondientes para dar solución a las historias de usuario y se realizaron las pruebas de aceptación que le propician al cliente conformidad y una mayor seguridad ante el sistema. Con el fin de este capítulo se da por terminada la propuesta final del sistema.

CONCLUSIONES

En el presente trabajo se realizó un estudio que permitió tener un conocimiento de la situación actual existente en la facultad 8 de la Universidad de las Ciencias Informáticas demostrándose así la necesidad de desarrollar un sistema que fuese capaz de automatizar los procesos referente a la labor llevada a cabo por los profesores guías en dicha facultad.

Con este trabajo se ofrece una solución utilizando el CMS Drupal acompañado de la metodología ágil XP. El desarrollo con este tipo de metodología permitió documentar el mismo desde el comienzo lo que facilitará una rápida comprensión y estudio. El desarrollo sobre Drupal permitirá que sea altamente configurable y adaptable a las necesidades, ya que las funcionalidades fueron desarrolladas de forma modular permitiendo una mejor portabilidad del sistema.

Como resultado se presenta una aplicación web en la cual se gestionara toda la información que llevan los profesores guías referente a los estudiantes de la facultad 8. La versión de este sistema fue validada y comprobada, demostrando así su utilidad y factibilidad.

RECOMENDACIONES

Junto al proceso investigativo que se llevó a cabo surgieron algunas ideas que serían recomendables tener en cuenta para futuras mejoras del sistema, a continuación se listan las mismas:

1. Agregar nuevas funcionalidades de acuerdo a las expectativas que surjan en la facultad 8 de la Universidad de las Ciencias Informáticas.
2. Escoger siempre que las condiciones lo permitan como línea de desarrollo la construcción de sistemas de gestión utilizando el CMS Drupal y la metodología XP.

REFERENCIAS BIBLIOGRÁFICAS

- ALBERTO, N. C. *MySQL 5.0 Reference Manual*, 2010. [15 de febrero del 2010, 8:45 am]. Disponible en: <http://dev.mysql.com/doc/refman/5.0/es/index.html>
- ATRIOWEB, E. *Sistemas de Gestión de Contenidos*, 15 de febrero del 2010, a las 2:15 pm]. Disponible en: <http://www.atrioweb.com/cms/>
- CERONJC, I. *Joomla CMS - Sistema de Gestión de Contenidos*, 2009. [15 de febrero del 2010, a las 9:00 pm]. Disponible en: http://www.ceronjc.com/index.php?option=com_content&view=article&id=60&Itemid=69
- DAMIÁN, P. V. *Los diferentes lenguajes de programación para la web*, 2 de Noviembre del 2007. [17 de febrero del 2010, 11:20 am]. Disponible en: <http://www.maestrosdelweb.com/principiantes/los-diferentes-lenguajes-de-programacion-para-la-web/>
- DANIEL, P. *PostGreSQL vs. MySQL*, 15 de febrero del 2010, 10:15 am]. Disponible en: [+http://danielpecos.com/docs/mysql_postgres/x15.html#AEN17](http://danielpecos.com/docs/mysql_postgres/x15.html#AEN17)
- FERNÁNDEZ, A. *Desarrollo Web con Software Libre*, 2006. [14 de enero del 2010, 10:20 am]. Disponible en: <http://www.webmasterlibre.com/>
- HERNÁNDEZ, A. *Seguridad en Internet*, 17 de enero del 2010, 2:00 pm]. Disponible en: http://www.capacinet.gob.mx/wb2/eMex/eMex_Glosario_de_terminos_Seguridad?page=29
- JOSE, R. A. *Características de Drupal*, Julio 17 del 2005 [16 de febrero del 2010, 9:15 am]. Disponible en: <http://drupal.org.es/caracteristicas>
- SÁNCHEZ, J. *¿Qué es un framework?*, 2007. [14 de enero del 2010, 11:00 am]. Disponible en: <http://jordisan.net/blog/2006/que-es-un-framework>
- CALDERÓN, AMARO. 2007. *Metodologías Ágiles*. Perú: s.n., 2007.
- RODRIGUEZ, MAITE, ORDOÑES, MEYLIN. 2007. Tesis LA METODOLOGÍA XP APLICABLE AL DESARROLLO DEL SOFTWARE EDUCATIVO EN CUBA. [26 de abril del 2010 hora 3:33 pm]
- SLIDESHARE INC. 2009. *slideshare*. [En línea] 2009. [20 de febrero de 2010.] Disponible en: http://www.slideshare.net/vivi_jocadi/rational-rose.

CARVAJAL, A. A. *Lenguaje Unificado de Modelado*. [En línea] 2005 [21 de enero de 2010]. Disponible en:

<http://www.google.com.cu/url?sa=t&source=web&ct=res&cd=4&ved=0CBMQFjAD&url=http%3A%2F%2Fsyacomputadores.googlepages.com%2FLenguajeUnificadodeModeladoUML.doc&rct=j&q=rational+rose+herramienta+de+modelado+caracteristicas&ei=4dpS7aZFsue8AaYvfSrBw&usg=AFQjCNG71V1H8qOzlxVxWm1kgFt2MCmlg>

ALTAMIRANO, A. V. *Comparativo de Entornos de Desarrollo Integrados* [21 de enero de 2010].

Disponible en:

<http://ubicuos.com/wp-content/uploads/2009/05/comparativoides.pdf>

INFO&SOFT. 2008. Info&Soft soluciones informáticas. [En línea] 2008. [25 de febrero de 2010.]

Disponible en:

http://infosoftmontero.blogspot.com/2008_02_01_archive.html.

Aulke, Gaylord. 2007. Zend Studio y Zend Platform en un. [En línea] 2007. [3 de marzo de 2010.]

BIBLIOGRAFÍA

1. **Beck, K.** Extreme Programming Explained: Addison Wesley, 2000.
2. **Beck, K. y Fowler, M.** Planeando en Programación Extrema. 2000.
3. **Shore, James y Warden, Shane.** The Art of Agile Development. : O'Reilly, 2007.
4. **Jeffries, Ron, Anderson, Ann y Hendrickson, Chet.** Extreme Programming Installed. : Addison Wesley, 2000.
5. **Amaro Calderón, Sarah Dámaris, Valverde Rebaza y Jorge Carlos.** Metodologías Ágiles. Perú, 2007.
6. **Barchini, Graciela Elisa.** *Métodos "I + D" de la Informática.*
7. **Camilo Javier Solis Álvarez y Roberth Gustavo Figueroa Díaz.** *Metodologías Tradicionales vs. Metodologías Ágiles.*
8. **Grady Booch, Robert C, Martin y James Newkirk.** *Oriented Analysis and Design with Applications.* s.l. : Addison Wesley Longman, 1998.
9. **José H. Canós, Patricio Letelier y M^a Carmen Penadés.** *Metodologías Ágiles en el Desarrollo de Software.*
10. **VanDyk, John K. 2008.** *Pro Drupal Development.* 2008.
11. **Stephens, Matt y Rosenberg, Doug.** Extreme Programming Refactored: The Case Against XP. Apress, 2003.
12. **Jorge Ferrer Zarzuela.** Metodologías Ágiles.
13. **Joskowicz.** Reglas y Prácticas en eXtreme Programming. 2008
14. **Cornejo, José Enrique González.** 2007. DocIRIS. DocIRIS. [Online] 2007.<http://www.docirs.cl/uml.htm>.
15. **CYTA.** 2006. [Online] 01 15, 2006.
<http://www.cyta.com.ar/biblioteca/bddoc/bdlibros/proyectoinformatico/libro/c5/c5.htm>.
16. **Letelier, Patricio.** 2006. CyTA. [Online] 01 15, 2006.
<http://www.cyta.com.ar/ta0502/v5n2a1.htm>.
17. **MYSQL.** 2008. Dev.mysql.com. [Online] 2008.
<http://dev.mysql.com/doc/refman/5.0/es/features.html>.

18. **Pecos, Daniel.** 2007. Netpecos. [Online] 2007
http://www.netpecos.org/docs/mysql_postgres/index.html.
19. AJAX un nuevo acercamiento a aplicaciones web. [En línea]
<http://www.uberbin.net/archivos/internet/ajax-un-nuevo-acercamiento-a-aplicaciones-web.php>.
20. **Bakken, Sæther, Schmid, Stig. y Egon.** Manual de Php. PHP Documentation Group. [En línea] 2003. <http://www.php.net/docs.php>.
21. **Sistema de gestión de Contenidos: Artículo: "CMS Quick Guide".** [En línea]
<http://www.opensourcecms.com/index.php?option=content&task=view&id=388>.
22. **Sistemas de Gestión de Contenidos: Tutorial: "CMS Tutorial".** [En línea]
[Citado el: 02 de febrero de 2009.]
<http://www.opensourcecms.com/index.php?option=content&task=view&id=500>.
23. **Sistemas de Gestión de Contenidos: "About". Drupal.org.** [En línea]
<http://drupal.org/features>.
24. **Sistemas de Gestión de Contenidos: "Module developer's guide". Drupal.org.** [En línea] <http://drupal.org/node/508>.
25. **Sistemas de Gestión de Contenidos: Artículo: "Introducción a los Sistemas de Gestión de Contenidos (CMS)".** [En línea]
<http://mosaic.uoc.edu/articulos/cms1204.html>.
26. **PostgreSQL.**[En Línea] <http://www.postgresql.org>
27. **PostgreSQL. Características fundamentales** [En Línea].
http://www.netpecos.org/docs/mysql_postgres/x15.html
28. **PostgreSQL Comparativas.**[En Línea]
http://www.netpecos.org/docs/mysql_postgres/x108.html

GLOSARIO DE TÉRMINOS

Akademós: Sitio en el cual se gestiona toda la información docente de los estudiantes, sus notas así como las planillas de matrícula. Dirección del sitio akademos.uci.cu

API: Del inglés Application Programming Interface - Interfaz de Programación de Aplicaciones es el conjunto de funciones y procedimientos (o métodos si se refiere a programación orientada a objetos) que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción.

Sockets: Número de identificación compuesto por dos números: La dirección IP y el número de puerto TCP. En la misma red, el nº IP es el mismo, mientras el nº de puerto es el que varía.

Unix: Es una familia de sistemas operativos libres. Soporta gran número de usuarios y posibilita la ejecución de distintas tareas de forma simultánea (multiusuario y multitarea).

Rollback's: Revertir una transacción. Por ejemplo, en una Base de Datos, una transacción se compone de una o más (un conjunto) de operaciones (insertar, eliminar, actualizar, etc.). Se inicia la transacción con algo como "Begin Transaction" y se cierra con "End Transaction". Cuando se quiere revertir el resultado de dichas operaciones (de una transacción), entonces se ejecuta el comando Rollback.

XHTML: Acrónimo de Extensible HyperText Markup Language (Lenguaje de Marcado de Hipertexto Extensible). Clase de especificaciones diseñada para que HTML sea conforme con el formato XML.

Historias de Usuario

Tabla 2.14 HU Consultar perfil de usuario.

Historia de Usuario	
No. : 2.	Nombre: Consultar perfil de usuario.
Usuario: Todos.	
Prioridad en Negocio (Alta / Media / Baja): Media.	Puntos Estimados:
Riesgos en Desarrollo (Alta / Media / Baja): Media.	Iteración Asignada: 2.
Descripción: Se brinda la posibilidad de ver los datos que el usuario ha llenado en la creación de su cuenta, además se le dará la opción de buscar otros perfiles de usuarios.	
Observaciones:	

Tabla 2.15 HU Consultar Grupos.

Historia de Usuario	
No. : 7	Nombre: Consultar Grupos
Usuario: todos.	
Prioridad en Negocio (Alta / Media / Baja): Media	Puntos Estimados:
Riesgos en Desarrollo (Alta / Media / Baja): Baja.	Iteración Asignada: 2
Descripción: Se brinda la opción de que todos los usuarios puedan ver todos los grupos que están creados.	
Observaciones:	

Tabla 2.16 HU Consultar años.

Historia de Usuario	
No. : 10	Nombre: Consultar años
Usuario: Administrador	
Prioridad en Negocio (Alta / Media / Baja): Media.	Puntos Estimados:
Riesgos en Desarrollo (Alta / Media / Baja): Baja.	Iteración Asignada: 2
Descripción: Se podrá consultar los años, el administrador podrá relacionar estos con los grupos docentes.	
Observaciones:	

Tabla 2.17 HU Consultar curso.

Historia de Usuario	
No. : 12	Nombre: Consultar curso.
Usuario: Todos.	
Prioridad en Negocio (Alta / Media / Baja): Media.	Puntos Estimados:
Riesgos en Desarrollo (Alta / Media / Baja): Media.	Iteración Asignada: 2
Descripción: Se brinda la posibilidad de consultar los cursos y los estudiantes y profesores que perteneces o pertenecieron a ellos.	
Observaciones:	

Tabla 2.18 HU Gestionar persona.

Historia de Usuario	
No. : 17	Nombre: Gestionar persona
Usuario: Administrador	
Prioridad en Negocio (Alta / Media / Baja): Media.	Puntos Estimados:

Riesgos en Desarrollo (Alta / Media / Baja): Baja.	Iteración Asignada: 2
Descripción: Brinda la posibilidad al usuario con el rol de Administrador de cambiar o introducir los usuarios del sistema (estudiantes, profesores, profesores guías, coordinadores de año).	
Observaciones:	

Tabla 2.19 HU Consultar persona.

Historia de Usuario	
No. : 18	Nombre: Consultar persona
Usuario: Todos	
Prioridad en Negocio (Alta / Media / Baja): Baja.	Puntos Estimados:
Riesgos en Desarrollo (Alta / Media / Baja): Baja.	Iteración Asignada: 3
Descripción: Brinda la posibilidad a todos los usuarios de consultar los usuarios registrados en el sistema.	
Observaciones:	

Tabla 2.20 HU Vincular grupo a un curso en un semestre.

Historia de Usuario	
No. : 19	Nombre: Vincular grupo a un curso en un semestre
Usuario: Dirección.	
Prioridad en Negocio (Alta / Media / Baja): Media.	Puntos Estimados:
Riesgos en Desarrollo (Alta / Media / Baja): Media.	Iteración Asignada: 2
Descripción: Se le brinda la posibilidad al usuario con el rol de Dirección de vincular a una etapa los grupos que le corresponden.	
Observaciones:	

Tabla 2.21 HU Vincular profesor guía a un grupo.

Historia de Usuario	
No. : 20	Nombre: Vincular profesor guía a un grupo
Usuario: Dirección.	
Prioridad en Negocio (Alta / Media / Baja): Media.	Puntos Estimados:
Riesgos en Desarrollo (Alta / Media / Baja): Media.	Iteración Asignada: 2
Descripción: Se le brinda la posibilidad al usuario con el rol de Dirección de vincular a un grupo el profesor guía que le corresponde.	
Observaciones:	

Tabla 2.22 HU Vincular estudiantes al grupo.

Historia de Usuario	
No. : 21	Nombre: Vincular estudiantes al grupo
Usuario: Dirección.	
Prioridad en Negocio (Alta / Media / Baja): Media.	Puntos Estimados:
Riesgos en Desarrollo (Alta / Media / Baja): Media.	Iteración Asignada: 2
Descripción: Se le brinda la posibilidad al usuario con el rol de Dirección de vincular los estudiantes al grupo que le corresponde.	
Observaciones:	

Tabla 2.23 HU Gestionar vínculos.

Historia de Usuario	
No. : 22	Nombre: Gestionar vínculos
Usuario: Administrador.	
Prioridad en Negocio (Alta / Media / Baja): Baja.	Puntos Estimados:

Riesgos en Desarrollo (Alta / Media / Baja): Media.	Iteración Asignada: 3
Descripción: Se permitirá al usuario con el rol de Administrador insertar y eliminar vínculos a otros sitios de interés.	
Observaciones:	

Tabla 2.24 HU Consultar vínculos.

Historia de Usuario	
No. :23	Nombre: Consultar vínculos
Usuario: Todos.	
Prioridad en Negocio (Alta / Media / Baja): Baja.	Puntos Estimados:
Riesgos en Desarrollo (Alta / Media / Baja): Baja.	Iteración Asignada: 3
Descripción: Se permitirá a todos los usuarios acceder a los vínculos a otros sitios.	
Observaciones:	

Tabla 2.25 HU Gestionar avisos.

Historia de Usuario	
No. :24	Nombre: Gestionar avisos
Usuario: Dirección.	
Prioridad en Negocio (Alta / Media / Baja): Baja.	Puntos Estimados:
Riesgos en Desarrollo (Alta / Media / Baja): Media.	Iteración Asignada: 3
Descripción: Se permitirá al usuario con el rol de dirección de crear, modificar y eliminar avisos y noticias de interés para los estudiantes y profesores.	
Observaciones:	

Tabla 2.26 HU Consultar avisos.

Historia de Usuario	
No. :25	Nombre: Consultar avisos

Usuario: Todos.	
Prioridad en Negocio (Alta / Media / Baja): Baja.	Puntos Estimados:
Riesgos en Desarrollo (Alta / Media / Baja): Media.	Iteración Asignada: 3
Descripción: Se brinda la posibilidad de que todos los usuarios puedan ver las noticias y avisos publicados.	
Observaciones:	

Tabla 2.27 HU Gestionar noticias.

Historia de Usuario	
No. : 26	Nombre: Gestionar noticias
Usuario: Todos.	
Prioridad en Negocio (Alta / Media / Baja): Baja.	Puntos Estimados:
Riesgos en Desarrollo (Alta / Media / Baja): Media.	Iteración Asignada: 3
Descripción: Se brinda la posibilidad de que todos los usuarios puedan ver las noticias y avisos publicados.	
Observaciones:	

Tabla 2.28 HU Consultar noticias.

Historia de Usuario	
No. : 27	Nombre: Consultar noticias
Usuario: Todos.	
Prioridad en Negocio (Alta / Media / Baja): Baja.	Puntos Estimados:
Riesgos en Desarrollo (Alta / Media / Baja): Media.	Iteración Asignada: 3
Descripción: Se brinda la posibilidad de que todos los usuarios puedan ver las noticias y avisos publicados.	
Observaciones:	

Tareas realizadas por cada módulo en la primera iteración

Módulo Procedencia

Tabla 3.16 Tarea 1 del módulo Procedencia.

Tarea	
Número de tarea: 1	Número de HU: 15
Nombre de la Tarea: Gestionar la procedencia de los estudiantes.	
Tipo de Tarea: Desarrollo	Puntos de estimación: 1
Fecha de inicio: 3 de Marzo del 2010	Fecha fin: 6 de Marzo del 2010
Programador responsable: Danais Rojas Suárez	
Descripción: Se crearan las funcionalidades para la gestión apropiada de la información referente a la procedencia de un estudiante. Se implementará el módulo procedencia.	

Tabla 3.17 Tarea 2 del módulo Procedencia.

Tarea	
Número de tarea: 2	Número de HU: 15
Nombre de la Tarea: Gestionar la procedencia de los estudiantes.	
Tipo de Tarea: Desarrollo	Puntos de estimación: 1
Fecha de inicio: 7 de Marzo del 2010	Fecha fin: 9 de Marzo del 2010
Programador responsable: Danais Rojas Suárez	
Descripción: Se crearan los permisos necesarios por roles para acceder a este módulo.	

Módulo Grupo

Tabla 3.18 Tarea 1 del módulo Grupo.

Tarea	
Número de tarea: 1	Número de HU: 6
Nombre de la Tarea: Configuración del módulo grupo	
Tipo de Tarea: Desarrollo	Puntos de estimación: 0.5

Fecha de inicio: 10 de Marzo del 2010	Fecha fin: 14 de Marzo del 2010
Programador responsable: Danais Rojas Suarez	
Descripción: Se implementará el módulo grupo.	

Módulo Proyecto_Educativo

Tabla 3.19 Tarea 1 del módulo Proyecto_Educativo.

Tarea	
Número de tarea: 1	Número de HU: 4
Nombre de la Tarea: Recolectar información referente a los proyectos educativos.	
Tipo de Tarea: Investigación	Puntos de estimación:
Fecha de inicio: 15 de Marzo del 2010	Fecha fin: 16 de Marzo del 2010
Programador responsable: Danais Rojas Suárez	
Descripción: Se recopilara información referente a los tipos de proyectos educativos que existen en la facultad.	

Tabla 3.20 Tarea 2 del módulo Proyecto_Educativo.

Tarea	
Número de tarea: 2	Número de HU: 4
Nombre de la Tarea: Gestionar proyectos educativos	
Tipo de Tarea: Desarrollo	Puntos de estimación: 0.5
Fecha de inicio: 17 de Marzo del 2010	Fecha fin: 19 de Marzo del 2010
Programador responsable: Danais Rojas Suárez	
Descripción: Se implementara el módulo proyecto educativo, definiendo cuantos tipos existen para una mejor implementación.	

Tabla 3.21 Tarea 3 del módulo Proyecto_Educativo.

Tarea	
Número de tarea: 3	Número de HU: 4
Nombre de la Tarea: Gestionar proyectos educativos	

Tipo de Tarea: Configuración	Puntos de estimación: 0.5
Fecha de inicio: 20 de Marzo del 2010	Fecha fin: 21 de Marzo del 2010
Programador responsable: Danais Rojas Suárez	
Descripción: Se configuraran los permisos por roles a este módulo.	

Módulo Año

Tabla 3.22 Tarea 1 del módulo Año.

Tarea	
Número de tarea: 1	Número de HU: 9
Nombre de la Tarea: Creación del módulo año.	
Tipo de Tarea: Desarrollo	Puntos de estimación: 0.5
Fecha de inicio: 22 de Marzo del 2010	Fecha fin: 23 de Marzo del 2010
Programador responsable: Danais Rojas Suárez	
Descripción: Se implementaran las funcionalidades y las tablas necesarias para poder gestionar el módulo año.	

Módulo Descarga

Tabla 3.23 Tarea 1 del módulo Descarga.

Tarea	
Número de tarea: 1	Número de HU: 8
Nombre de la Tarea: Gestionar descargas.	
Tipo de Tarea: Desarrollo.	Puntos de estimación: 0.3
Fecha de inicio: 24 de Marzo del 2010	Fecha fin: 25 de Marzo del 2010
Programador responsable: Danais Rojas Suárez	
Descripción: Se implementaran las funcionalidades de descargas para el módulo proyecto educativo, además se implementaran las opciones de descargas y subir archivos al sistema por el administrador.	

Tabla 3.24 Tarea 2 del módulo Descarga.

Tarea

Número de tarea: 2	Número de HU: 8
Nombre de la Tarea: Gestionar descargas.	
Tipo de Tarea: Configuración.	Puntos de estimación: 0.3
Fecha de inicio: 26 de Marzo del 2010	Fecha fin: 27 de Marzo del 2010
Programador responsable: Danais Rojas Suárez	
Descripción: Se especificaran los permisos necesarios para permitir la descarga y subida de archivos e dependencia del tipo de usuario que sea.	

Módulo Curso

Tabla 3.25 Tarea 1 del módulo Curso.

Tarea	
Número de tarea: 1	Número de HU: 11
Nombre de la Tarea: Configuración del módulo curso.	
Tipo de Tarea: Desarrollo	Puntos de estimación: 0.5
Fecha de inicio: 27 de Marzo del 2010	Fecha fin: 28 de Marzo del 2010
Programador responsable: Danais Rojas Suárez	
Descripción: Se implementarán las tablas y los archivos necesarios para poder configurar correctamente el módulo curso.	

Módulo Departamento

Tabla 3.26 Tarea 1 del módulo Departamento.

Tarea	
Número de tarea: 1	Número de HU: 13
Nombre de la Tarea: Gestionar módulo departamento.	
Tipo de Tarea: Investigación.	Puntos de estimación:
Fecha de inicio: 29 de Marzo del 2010	Fecha fin: 30 de Marzo del 2010
Programador responsable: Danais Rojas Suárez	
Descripción: Se investigaran los diferentes departamentos existentes en la	

facultad 8.

Tabla 3.27 Tarea 2 del módulo Departamento.

Tarea	
Número de tarea: 2	Número de HU: 13
Nombre de la Tarea: Gestionar módulo departamento.	
Tipo de Tarea: Desarrollo	Puntos de estimación: 1
Fecha de inicio: 31 de Marzo del 2010	Fecha fin: 2 de Abril del 2010
Programador responsable: Danais Rojas Suárez	
Descripción: Se implementará el módulo departamento.	

Tabla 3.28 Tarea 3 del módulo Departamento.

Tarea	
Número de tarea: 3	Número de HU: 13
Nombre de la Tarea: Gestionar módulo departamento.	
Tipo de Tarea: Configuración	Puntos de estimación: 1
Fecha de inicio: 3 de Abril del 2010	Fecha fin: 4 de Abril del 2010
Programador responsable: Danais Rojas Suárez	
Descripción: Se configuraran los permisos a los roles que podrán tener acceso a este módulo.	

Tareas realizadas por cada módulo en la segunda iteración

Módulo Persona

Tabla 3.34 Tarea 4 del módulo Persona.

Tarea	
Número de tarea: 4	Número de HU: 2
Nombre de la Tarea: Consulta del perfil de usuario.	
Tipo de Tarea: Configuración.	Puntos de estimación: 0.5
Fecha de inicio: 18 de Abril del 2010	Fecha fin: 19 de Abril del 2010
Programador responsable: Danais Rojas Suarez	

Descripción: Se establecerán los permisos a los roles que tendrán acceso a la consulta de los perfiles de usuarios creados.

Tabla 3.35 Tarea 5 del módulo Persona.

Tarea	
Número de tarea: 5	Número de HU: 17
Nombre de la Tarea: Gestionar persona.	
Tipo de Tarea: Configuración.	Puntos de estimación: 1
Fecha de inicio: 20 de Abril del 2010	Fecha fin: 21 de Abril del 2010
Programador responsable: Danais Rojas Suarez	
Descripción: Se configurara el módulo persona, permitiendo la inserción de los diferentes usuarios del sistema (estudiantes, profesores, profesores guías, coordinadores de año).	

Módulo Curso

Tabla 3.36 Tarea 2 del módulo Curso.

Tarea	
Número de tarea: 2	Número de HU: 12
Nombre de la Tarea: Configuración de los permisos al módulo curso.	
Tipo de Tarea: Configuración	Puntos de estimación: 0.5
Fecha de inicio: 22 de Abril del 2010	Fecha fin: 23 de Abril del 2010
Programador responsable: Danais Rojas Suárez	
Descripción: Se configuraran los permisos a los roles que tendrán acceso a este módulo.	

Módulo Año

Tabla 3.37 Tarea 2 del módulo Año.

Tarea

Número de tarea: 2	Número de HU: 10
Nombre de la Tarea: Consultar módulo año.	
Tipo de Tarea: Configuración	Puntos de estimación: 0.5
Fecha de inicio: 24 de Abril del 2010	Fecha fin: 25 de Abril del 2010
Programador responsable: Danais Rojas Suárez	
Descripción: Se establecerán los permisos correspondientes para los roles que puedan acceder al módulo año.	

Tareas realizadas por cada módulo en la segunda iteración.

Módulo Noticias

Tabla 3.41 Tarea 1 del módulo Noticias.

Tarea	
Número de tarea: 1	Número de HU: 26
Nombre de la Tarea: Creación del módulo noticias.	
Tipo de Tarea: Desarrollo	Puntos de estimación: 1
Fecha de inicio: 28 de Abril del 2010	Fecha fin: 29 de Abril del 2010
Programador responsable: Danais Rojas Suárez	
Descripción: Se implementará el módulo noticias.	

Tabla 3.42 Tarea 2 del módulo Noticias.

Tarea	
Número de tarea: 2	Número de HU: 27
Nombre de la Tarea: Configuración del módulo noticias.	
Tipo de Tarea: Configuración	Puntos de estimación: 0.3
Fecha de inicio: 29 de Abril del 2010	Fecha fin: 30 de Abril del 2010
Programador responsable: Danais Rojas Suárez	
Descripción: Se configurarán los permisos a los diferentes roles, para determinar quien tiene acceso a creas noticias y quien tiene acceso a consultar las noticias.	

Módulo Avisos

Tabla 3.43 Tarea 1 del módulo Avisos.

Tarea	
Número de tarea: 1	Número de HU: 24
Nombre de la Tarea: Gestionar avisos.	
Tipo de Tarea: Desarrollo	Puntos de estimación: 1.2
Fecha de inicio: 2 de Mayo del 2010	Fecha fin: 3 de Mayo del 2010
Programador responsable: Danais Rojas Suárez	
Descripción: Se llevará a cabo la implementación del módulo avisos, permitiendo la inserción, modificación y eliminación de estos.	

Tabla 3.44 Tarea 2 del módulo Avisos.

Tarea	
Número de tarea: 2	Número de HU: 25
Nombre de la Tarea: Configuración del módulo avisos	
Tipo de Tarea: Desarrollo	Puntos de estimación: 0.4
Fecha de inicio: 3 de Mayo del 2010	Fecha fin: 4 de Mayo del 2010
Programador responsable: Danais Rojas Suárez	
Descripción: Se establecerán los permisos a los diferentes roles especificando cuales pueden crear avisos y quienes solo pueden consultarlos.	

Módulo Persona

Tabla 3.45 Tarea 6 del módulo Persona.

Tarea	
Número de tarea: 6	Número de HU: 18
Nombre de la Tarea: Consultar persona.	
Tipo de Tarea: Configuración	Puntos de estimación: 2

Fecha de inicio: 4 de Mayo del 2010	Fecha fin: 6 de Mayo del 2010
Programador responsable: Danais Rojas Suárez	
Descripción: Se configurará el módulo persona permitiendo listar las personas registradas y que se muestre todos sus datos.	

Pruebas

Tabla 3.49 Prueba 1 al módulo curso.

Caso de prueba de aceptación	
Código: HU11_P1	Historia de usuario: 11
Nombre: Gestionar curso	
Descripción: La aplicación debe brindar la opción de insertar, modificar y eliminar toda la información referente a los cursos.	
Condiciones de ejecución : El usuario debe estar autenticado El usuario debe pertenecer al rol de Dirección	
Entrada/ Pasos de ejecución: Se intenta insertar, modificar y eliminar un curso.	
Resultado esperado: Que se pueda gestionar un curso.	
Evaluación de la prueba: Prueba satisfactoria.	

Tabla 3.50 Prueba 2 al módulo curso.

Caso de prueba de aceptación	
Código: HU12_P1	Historia de usuario: 12
Nombre: Consultar curso	
Descripción: La aplicación debe brindar la opción de mostrar el listado de todos los cursos.	
Condiciones de ejecución : El usuario debe de estar autenticado Todos los usuarios deben tener acceso a esta función	

Entrada/ Pasos de ejecución : Se intenta consultar los cursos.
Resultado esperado: Que la aplicación muestre los cursos correctamente.
Evaluación de la prueba: Prueba satisfactoria.

Tabla 3.51 Prueba 1 al módulo grupo.

Caso de prueba de aceptación	
Código: HU6_P1	Historia de usuario: 6
Nombre: Gestionar grupo	
Descripción: La aplicación debe brindar la opción de gestionar grupos.	
Condiciones de ejecución : El usuario debe estar autenticado El usuario debe pertenecer al rol de Dirección.	
Entrada/ Pasos de ejecución : Se intenta insertar, modificar y eliminar un grupo.	
Resultado esperado: Que la aplicación inserte, modifique y elimine los grupos correctamente	
Evaluación de la prueba: Prueba satisfactoria.	

Tabla 3.52 Prueba 2 al módulo grupo.

Caso de prueba de aceptación	
Código: HU7_P1	Historia de usuario: 7
Nombre: Consultar grupo	
Descripción: La aplicación debe brindar la opción de consultar todos los grupos que se encuentran registrados.	
Condiciones de ejecución: El usuario debe estar autenticado	
Entrada/ Pasos de ejecución: Se intenta listar los grupos.	
Resultado esperado: Que se muestre una lista con todos los grupos que existen.	

Evaluación de la prueba: Prueba satisfactoria
--

Tabla 3.53 Prueba 3 al módulo grupo.

Caso de prueba de aceptación	
Código: HU19_P1	Historia de usuario: 19
Nombre: Vincular grupo a un curso en un semestre	
Descripción: La aplicación debe brindar la opción de poder vincular un grupo a un curso en un semestre determinado.	
Condiciones de ejecución : El usuario debe estar autenticado El usuario debe pertenecer al rol de Dirección.	
Entrada/ Pasos de ejecución: Se intenta vincular un grupo que ya existe a un curso específico en un semestre determinado.	
Resultado esperado: Que se pueda vincular satisfactoriamente un grupo a un curso en un semestre.	
Evaluación de la prueba: Prueba satisfactoria.	

Tabla 3.54 Prueba 4 al módulo grupo.

Caso de prueba de aceptación	
Código: HU20_P1	Historia de usuario: 20
Nombre: Vincular profesor guía al grupo	
Descripción: Prueba para la funcionalidad vincular profesor guía a un grupo.	
Condiciones de ejecución : El usuario debe estar autenticado El usuario debe pertenecer al rol de Dirección.	
Entrada/ Pasos de ejecución: Se intenta vincular un profesor ya registrado en la aplicación a un grupo.	
Resultado esperado: Que se logre vincular satisfactoriamente un profesor a un grupo, convirtiéndolo en su profesor guía.	
Evaluación de la prueba: Prueba satisfactoria.	

Tabla 3.55 Prueba 5 al módulo grupo.

Caso de prueba de aceptación	
Código: HU21_P1	Historia de usuario: 21
Nombre: Vincular estudiantes a un grupo	
Descripción: Prueba para la funcionalidad vincular los estudiantes a un grupo.	
Condiciones de ejecución : El usuario debe estar autenticado El usuario debe pertenecer al rol de Dirección	
Entrada/ Pasos de ejecución: Se intenta vincular los estudiantes registrados en la aplicación a un grupo en específico.	
Resultado esperado: Se debe poder vincular los estudiantes a los grupos correctamente.	
Evaluación de la prueba: Prueba satisfactoria.	

Tabla 3.56 Prueba 1 al módulo avisos.

Caso de prueba de aceptación	
Código: HU24_P1	Historia de usuario: 24
Nombre: Gestionar avisos	
Descripción: Prueba para poder gestionar avisos en el portal	
Condiciones de ejecución : El usuario debe estar autenticado El usuario debe pertenecer al rol de Dirección	
Entrada/ Pasos de ejecución: Se intenta insertar, modificar y eliminar avisos.	
Resultado esperado: La aplicación debe permitir poder gestionar los avisos satisfactoriamente.	
Evaluación de la prueba: Prueba satisfactoria.	

Tabla 3.57 Prueba 2 al módulo avisos.

Caso de prueba de aceptación	
Código: HU25_P1	Historia de usuario: 25
Nombre: Consultar avisos	
Descripción: Prueba a la funcionalidad consultar avisos.	
Condiciones de ejecución: El usuario debe estar autenticado.	
Entrada/ Pasos de ejecución: Se intenta listar y consultar los avisos existentes.	
Resultado esperado: Que se pueda listar correctamente los avisos.	
Evaluación de la prueba: Prueba satisfactoria.	

Tabla 3.58 Prueba 1 al módulo noticias.

Caso de prueba de aceptación	
Código: HU26_P1	Historia de usuario: 26
Nombre: Gestionar noticias	
Descripción: Prueba para gestionar las noticias	
Condiciones de ejecución: El usuario debe estar autenticado El usuario debe pertenecer al rol de Dirección	
Entrada/ Pasos de ejecución: Se intenta insertar, modificar y eliminar noticias en la aplicación.	
Resultado esperado: Que se pueda gestionar correctamente las noticias.	
Evaluación de la prueba: Prueba satisfactoria.	

Tabla 3.59 Prueba 2 al módulo noticias.

Caso de prueba de aceptación	
Código: HU27_P1	Historia de usuario: 27
Nombre: Consultar noticias	
Descripción: Prueba para la funcionalidad de consultar las noticias.	
Condiciones de ejecución: El usuario debe estar autenticado	

Entrada/ Pasos de ejecución: Se intenta listar y consultar las noticias.
Resultado esperado: Que se puedan listar las noticias y se puedan consultar.
Evaluación de la prueba: Prueba satisfactoria

Tabla 3.60 Prueba 1 al módulo descargas.

Caso de prueba de aceptación	
Código: HU8_P1	Historia de usuario: 8
Nombre: Gestionar descargas.	
Descripción: Prueba para gestionar las descargas.	
Condiciones de ejecución : El usuario debe estar autenticado El usuario debe pertenecer al rol de Dirección, Coordinador de año o Profesor guía.	
Entrada/ Pasos de ejecución: Se intentará subir un archivo al sistema y se pretenderá descargarlos de la aplicación.	
Resultado esperado: Que se pueda subir archivos y se puedan descargar correctamente.	
Evaluación de la prueba: Prueba satisfactoria.	

Tabla 3.61 Prueba 1 al módulo vínculo.

Caso de prueba de aceptación	
Código: HU22_P1	Historia de usuario: 22
Nombre: Gestionar vínculos	
Descripción: Prueba para adicionar, modificar y eliminar un vínculo.	
Condiciones de ejecución : El usuario debe estar autenticado El usuario debe pertenecer al rol Dirección.	
Entrada/ Pasos de ejecución: Se pretende insertar, modificar y eliminar un vínculo.	
Resultado esperado: Que se pueda gestionar satisfactoriamente los vínculos en la aplicación.	
Evaluación de la prueba: Prueba satisfactoria.	

Tabla 3.62 Prueba 2 al módulo vínculo.

Caso de prueba de aceptación	
Código: HU23_P1	Historia de usuario: 23
Nombre: Consultar vínculos	
Descripción: Prueba para listar los vínculos publicados.	
Condiciones de ejecución: El usuario tiene que estar autenticado.	
Entrada/ Pasos de ejecución: Se intentará listar y consultar todos los vínculos publicados.	
Resultado esperado: Que se muestre un listado de todos los vínculos publicados.	
Evaluación de la prueba: Prueba satisfactoria.	

Tabla 3.63 Prueba 1 al módulo año.

Caso de prueba de aceptación	
Código: HU9_P1	Historia de usuario: 9
Nombre: Gestionar año	
Descripción: Prueba para insertar, modificar y eliminar un año.	
Condiciones de ejecución : El usuario tiene que estar autenticado El usuario debe pertenecer al rol Dirección.	
Entrada/ Pasos de ejecución: Se pretende gestionar un año en la aplicación.	
Resultado esperado: Que se logre insertar, modificar y eliminar un año correctamente.	
Evaluación de la prueba: Prueba satisfactoria.	

Tabla 3.64 Prueba 2 al módulo año.

Caso de prueba de aceptación	
Código: HU10_P1	Historia de usuario: 10
Nombre: Consultar año.	
Descripción: Prueba para listar los años.	

Condiciones de ejecución: El usuario tiene que estar autenticado.
Entrada/ Pasos de ejecución: Se intenta listar todos los años.
Resultado esperado: Que se logre listar todos los años satisfactoriamente.
Evaluación de la prueba: Prueba satisfactoria.

Tabla 3.65 Prueba 1 al módulo departamento.

Caso de prueba de aceptación	
Código: HU13_P1	Historia de usuario: 13
Nombre: Gestionar departamento	
Descripción: Prueba para gestionar un departamento.	
Condiciones de ejecución : El usuario tiene que estar autenticado. El usuario debe pertenecer al rol de Dirección.	
Entrada/ Pasos de ejecución: Se pretende insertar, modificar, y eliminar diferentes departamentos.	
Resultado esperado: Que se logre gestionar los departamentos correctamente.	
Evaluación de la prueba: Prueba satisfactoria.	

Tabla 3.66 Prueba 2 al módulo departamento.

Caso de prueba de aceptación	
Código: HU14_P1	Historia de usuario: 14
Nombre: Consultar departamento.	
Descripción: Prueba para listar los departamentos.	
Condiciones de ejecución: El usuario debe estar autenticado.	
Entrada/ Pasos de ejecución: Se pretende listar todos los departamentos existentes en la aplicación.	
Resultado esperado: Que se listen satisfactoriamente los departamentos.	
Evaluación de la prueba: Prueba satisfactoria.	

Tabla 3.67 Prueba 1 al módulo procedencia.

Caso de prueba de aceptación	
Código: HU15_P1	Historia de usuario: 15
Nombre: Gestionar procedencia.	
Descripción: Prueba para insertar, modificar y eliminar la procedencia.	
Condiciones de ejecución : El usuario tiene que estar autenticado. El usuario debe pertenecer al rol de Dirección.	
Entrada/ Pasos de ejecución: Se pretende insertar, modificar, y eliminar la procedencia de los estudiantes.	
Resultado esperado: Que se logre gestionar correctamente la procedencia de los estudiantes.	
Evaluación de la prueba: Prueba satisfactoria.	

Tabla 3.68 Prueba 2 al módulo procedencia.

Caso de prueba de aceptación	
Código: HU16_P1	Historia de usuario: 16
Nombre: Consultar procedencia	
Descripción: Prueba para listar la procedencia de los estudiantes.	
Condiciones de ejecución: El usuario debe estar autenticado.	
Entrada/ Pasos de ejecución: Se intenta listar la procedencia de los estudiantes.	
Resultado esperado: Que la aplicación muestre un listado de la procedencia de los estudiantes.	
Evaluación de la prueba: Prueba satisfactoria.	

Tabla 3.69 Prueba 1 al módulo proyecto educativo.

Caso de prueba de aceptación	
Código: HU4_P1	Historia de usuario: 4
Nombre: Gestionar proyecto educativo.	

Descripción: Prueba para gestionar los diferentes proyectos educativos.
Condiciones de ejecución : El usuario tiene que estar autenticado. El usuario debe pertenecer a los roles de dirección, coordinador de año o profesor guía.
Entrada/ Pasos de ejecución: Se pretende insertar, modificar y eliminar los proyectos educativos.
Resultado esperado: Que se logre gestionar de manera correcta los proyectos educativos.
Evaluación de la prueba: Prueba satisfactoria.

Tabla 3.70 Prueba 2 al módulo proyecto educativo.

Caso de prueba de aceptación	
Código: HU5_P1	Historia de usuario: 5
Nombre: Consultar proyecto educativo.	
Descripción: Prueba para listar los proyectos educativos almacenados en la aplicación.	
Condiciones de ejecución: El usuario debe estar autenticado.	
Entrada/ Pasos de ejecución: Se pretende listar los proyectos educativos.	
Resultado esperado: Que se muestre una lista de los proyectos educativos divididos en su categoría.	
Evaluación de la prueba: Prueba satisfactoria.	