

Universidad de las Ciencias Informáticas

Facultad 6



Desarrollo del Portal Web de la Facultad 6.

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

Autores: Milenys Martínez Pérez

Liván Rodríguez Miranda

Tutores: MsC. Maikel Yelandi Leyva Vázquez

MsC. Olga Catalina Fontova De Los Reyes

Co-tutor: Lic. Leonardo Castillo Martínez

Ciudad de la Habana, Cuba.

Junio ,2008

“Año 50 de la Revolución.”



“Todos y cada uno de nosotros paga puntualmente su cuota de sacrificio conscientes de recibir el premio en la satisfacción del deber cumplido, conscientes de avanzar con todos hacia el Hombre Nuevo que se vislumbra en el horizonte.”

Ernesto Che Guevara.

DECLARACIÓN DE AUDITORÍA

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

Para que así conste firmo la presente a los 28 días del mes de junio del año 2008.

Milenys Martínez Pérez

MsC. Maikel Yelandi Leyva Vázquez

Firma del Autor

Firma del Tutor

Liván Rodríguez Miranda

MsC. Olga Catalina Fontova De Los Reyes

Firma del Autor

Firma del Tutor

Lic. Leonardo Castillo Martínez

Firma del Co-tutor

DATOS DE CONTACTO

Tutores

MsC. Maikel Yelandi Leyva Vázquez

Universidad de las Ciencias Informáticas, La Habana, Cuba.

Email: mleyvaz@uci.cu

MsC. Olga Catalina Fontova De Los Reyes

Universidad de las Ciencias Informáticas, La Habana, Cuba.

Email: fontova@uci.cu

Co-tutor

Lic. Leonardo Castillo Martínez

Universidad de las Ciencias Informáticas, La Habana, Cuba.

Email: leonardoc@uci.cu

AGRADECIMIENTOS

Ya han pasado cinco cursos de sacrificios, sueños, alegrías y ha llegado el momento de enfrentarnos al mundo como profesionales. Cada uno de nuestros logros trae consigo el amor, la comprensión y ayuda incondicional de todas aquellas personas que de una forma u otra fueron partícipes de que este momento existiese.

Un agradecimiento especial a nuestro Comandante en Jefe por darnos la oportunidad de estudiar en esta Universidad de Excelencia.

A nuestros padres por depositar toda su confianza en nosotros, apoyarnos y guiarnos en todo momento.

A nuestros abuelos por toda su paciencia y dedicación durante todos estos años.

A nuestros hermanos que siempre nos dieron aliento para seguir adelante.

A nuestra familia y a todas las personas que siempre nos han apoyado.

A nuestros tutores Maykel, Olga y Leonardo por guiarnos en esta investigación, por sus sugerencias siempre bien recibidas.

A Pacheco por su ayuda incondicional.

A nuestros amigos, a los viejos y a los nuevos, a los que ya no están, quienes nos ayudaron, apoyaron y aconsejaron en los momentos más difíciles, con los que compartimos una larga etapa de la vida.

A Derick, Arieskien, Martha y Anabel por toda su ayuda y apoyo, por estar siempre que los necesitamos.

A nuestros profesores por haber contribuido en nuestra formación como profesionales.

A todos muchas gracias.

DEDICATORIA

A mi mami Milagros por su cariño infinito, por ella y para ella todo mi esfuerzo y voluntad en estos 5 años. Porque este es también tu triunfo.

A mis abuelos Fela, Eddy y Enrique por su paciencia y preocupación constante.

A mi abuelo Pedro, donde quiera que esté.

A mi padrastro por todo su apoyo y su ayuda incondicional.

A mis hermanas Miledys, Maidelys y Mailenys por ser mi fuente de inspiración.

A Raúl por su apoyo constante y sus sabios consejos.

A mis compañeros y amigos por las experiencias compartidas y los momentos vividos, por todo su cariño, apoyo y aliento cuando más lo necesité.

A todas las personas que han pasado por mi vida dejando una huella indestructible.

Milenys Martínez Pérez

A mis padres Teresa y Luisito por enseñarme a luchar para alcanzar mis metas, por su amor y confianza en todo momento. ¡Mis logros son para ustedes!

A mis hermanos “Biembe” y “Tata” por confiar en mí y apoyarme en los momentos buenos y malos.

A toda mi familia que ha sabido apoyarme y entregarme su amor cada instante.

A mis amigos todos, a los de la UCI por compartir conmigo todos esos momentos de alegría y tristeza de 5 años y a los otros que se han preocupado por mí y me han sabido dar apoyo y cariño.

A todos aquellos que han creído en mí.

Liván Rodríguez Miranda

RESUMEN

La Facultad 6 de la Universidad de las Ciencias Informáticas necesita un control de toda la información que se maneja en las distintas áreas. En la actualidad cuenta con el Portal Web de la FEU, el cual brinda una serie de informaciones referentes a noticias relevantes, encuestas, foros de discusión y otros temas de interés; pero el mismo no muestra información relacionada con la producción, la investigación, la administración de la facultad, la docencia y temas típicos de cada departamento, por tal motivo se requiere de un Portal Web que integre todas estas áreas y permita mantener informada sistemáticamente a la facultad. Teniendo en cuenta esta necesidad, el objetivo fundamental del presente trabajo es desarrollar un Portal Web que permita mantener informada de forma sistemática a la comunidad universitaria de la Facultad 6 de la Universidad de las Ciencias Informáticas. El sistema fue llevado a cabo siguiendo los pasos que propone el Proceso Unificado de Desarrollo de Software y empleándose para el modelado del mismo la herramienta CASE Visual Paradigm.

PALABRAS CLAVES:

Portal Web, Facultad 6, Universidad de las Ciencias Informáticas.

TABLA DE CONTENIDOS

AGRADECIMIENTOS	II
DEDICATORIA.....	III
RESUMEN	IV
INTRODUCCIÓN	1
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA	4
1.1 Aplicaciones cliente-servidor. Los Portales Web.	4
1.1.1 Aplicaciones cliente-servidor.	4
1.1.2 Internet/Intranet: ¿Cuál es la diferencia?.....	4
1.1.3 Portales Web.	5
1.2 Sistema de Gestión de Contenido (CMS).	8
1.2.1 ¿Qué es un CMS?	8
1.2.2 CMS de código abierto y propietarios.....	9
1.2.3 Principales CMS de código abierto.....	10
1.3 Metodología utilizada, tendencia y tecnologías actuales.	13
1.3.1 Tecnología del lado del cliente.	13
1.3.2 Tecnología del lado del Servidor. PHP.....	14
1.3.3 Sistema de Gestión de Base de Datos.....	15
1.3.4 Servidor Web. Apache.....	17
1.4 Metodología de desarrollo.....	17
1.4.1 Metodología Extreme Programming (XP).....	17
1.4.2 Microsoft Solution Framework (MSF).	18
1.4.3 Rational Unified Process (RUP).	19
1.4.3.1 Flujos de Trabajo Definidos.....	21
1.5 Lenguaje de Modelado.....	22
1.6 Herramientas a utilizar.	23
1.6.1 Herramientas CASE.	23
1.6.2 Herramientas para el desarrollo de aplicaciones Web.	24
1.6.3 Herramientas para el trabajo con la Base de Datos.	26
1.6.4 Herramientas para la gestión de requisitos.	26
1.6.5 Ambiente de Desarrollo Integrado (IDE).....	27
1.7 Conclusiones.....	28
CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA.....	29
2.1 Modelo de Dominio.	29
2.1.1 Conceptos del Modelo de Dominio.....	31
2.1.2 Reglas del Negocio.....	32
2.2 Requisitos Funcionales.	32
2.3 Requisitos no Funcionales.	38
2.4 Descripción del Sistema Propuesto.	40
2.5 Definición de los Actores del Sistema.	40
2.6 Definición de los Casos de Uso del Sistema.....	41
2.6.1 Patrones de Casos de Uso.....	41

2.6.2 Listado de Casos de Uso.....	42
2.7 Diagrama de Casos de Uso del Sistema.	44
2.8 Descripción de los Casos de Uso del Sistema.....	49
2.9 Vista de Casos de Uso.....	53
2.10 Conclusiones.....	54
CAPÍTULO 3: DESCRIPCIÓN DE LA SOLUCIÓN PROPUESTA.....	55
3.1 Descripción de la Solución Propuesta.....	55
3.2 Arquitectura del Sistema.	57
3.3 Vista de Despliegue.	61
3.4 Prototipos de Interfaz de Usuario.....	62
3.5 Mapa de Navegación.	64
3.6 Modelo Físico.....	65
3.7 Principios de Protección y Seguridad.....	67
3.8 Conclusiones.....	67
CAPÍTULO 4: IMPLEMENTACIÓN.....	68
4.1 Diagrama de componentes.....	68
4.2 Código fuente de los principales componentes y descripciones de los mismos.....	72
4.3 Conclusiones.....	75
CONCLUSIONES.....	76
RECOMENDACIONES.....	77
REFERENCIAS BIBLIOGRÁFICAS.....	78
BIBLIOGRAFÍA.....	80
ANEXOS.....	83
GLOSARIO.....	95

ÍNDICE DE FIGURAS

Figura 1. Representación de la Metodología Extreme Programming.	17
Figura 2. Representación de la Metodología MSF.....	18
Figura 3. Representación de RUP en Dos Dimensiones.	19
Figura 4. Diagrama de clases del Modelo de Dominio.....	30
Figura 5. Diagrama de CUS General.	45
Figura 6. Diagrama de CUS correspondiente al paquete Gestión de Contenido.....	46
Figura 7. Diagrama de CUS correspondiente al paquete Administración del Sistema.	46
Figura 8. Diagrama de CUS correspondiente al paquete Producción.	47
Figura 9. Diagrama de CUS correspondiente al paquete Extensión Universitaria.....	47
Figura 10. Diagrama de CUS correspondiente al paquete Tesis.....	48
Figura 11. Diagrama de CUS correspondiente al paquete Horario-Máquina de Proyectos.....	48
Figura 12. Casos de uso arquitectónicamente significativos.	53
Figura 13. Representación del patrón MVC.	58
Figura 14. Representación del patrón MVC en Drupal.	59
Figura 15. Representación del patrón 3 Capas.....	60
Figura 16. Representación de la integración de MVC y 3 Capas.	61
Figura 17. Diagrama de Despliegue.....	62
Figura 18. Prototipo no funcional CU- Visualizar Datos Generales: Escenario Visualizar Estudiantes.	62
Figura 19. Prototipo no funcional CU- Adicionar Proyecto.....	63
Figura 20. Prototipo no funcional CU- Editar Proyecto.....	63
Figura 21. Prototipo no funcional CU- Asignar Profesor- Proyecto: Escenario Asignar Proyecto.	63
Figura 22. Mapa de Navegación correspondiente a Profesor-Estudiante.....	64
Figura 23. Mapa de Navegación correspondiente a Extensión Universitaria.....	64
Figura 24. Mapa de Navegación correspondiente a Horario-Máquina.....	65
Figura 25. Modelo Físico.....	66
Figura 26. Diagrama de Componentes CU-Visualizar Datos Generales.	69
Figura 27. Diagrama de Componentes CU-Adicionar Proyecto.....	70
Figura 28. Diagrama de Componentes CU-Editar Proyecto	70
Figura 29. Diagrama de Componentes CU-Asignar Profesor- Proyecto.....	71
Figura 30. Diagrama de Componentes CU-Eliminar Profesor- Proyecto.....	71
Figura 31. Diagrama de Componentes CU-Asignar Estudiante- Proyecto.	72
Figura 32. Diagrama de Componentes CU-Eliminar Estudiante- Proyecto.	72
Figura 33. Código correspondiente al fichero “config.php” del CU Adicionar Tesis.....	73
Figura 34. Código correspondiente al fichero “listar_lectura.php” del CU Visualizar Datos Generales: Escenario Visualizar Estudiantes.	74
Figura 35. Representación de la arquitectura Cliente-Servidor.	83

ÍNDICE DE TABLAS

Tabla 1. Descripción de los Actores del Sistema.	41
Tabla 2. Descripción del caso de uso Visualizar Datos Generales.	49
Tabla 3. Descripción del caso de uso Adicionar Proyecto.	50
Tabla 4. Descripción del caso de uso Editar Proyecto.	50
Tabla 5. Descripción del caso de uso Asignar Profesor-Proyecto. (Descripción ampliada: Ver Anexo 2)	51
Tabla 6. Descripción del caso de uso Eliminar Profesor-Proyecto. (Descripción ampliada: Ver Anexo 2)	51
Tabla 7. Descripción del caso de uso Asignar Estudiante-Proyecto. (Descripción ampliada: Ver Anexo 2)	52
Tabla 8. Descripción del caso de uso Eliminar Estudiante-Proyecto. (Descripción ampliada: Ver Anexo 2)	53
Tabla 9. Descripción ampliada del caso de uso Asignar Profesor-Proyecto.....	83
Tabla 10. Descripción ampliada del caso de uso Eliminar Profesor-Proyecto.....	86
Tabla 11. Descripción ampliada del caso de uso Asignar Estudiante-Proyecto.	88
Tabla 12. Descripción ampliada del caso de uso Eliminar Estudiante-Proyecto.	92

INTRODUCCIÓN

En esta era de grandes avances tecnológicos, las corporaciones utilizan con más frecuencia las ventajas que ofrece la tecnología para mantenerse a la vanguardia y obtener así una ventaja competitiva. A mediados de la década de los noventa, el uso del Web browser que permite navegar por Internet de una forma gráfica, revoluciona y crea un precedente en el uso de las comunicaciones. Este nuevo paradigma abrió las puertas a nuevos estilos y formas de comunicarse y trabajar. Una de estas formas es el Portal Web que empezó a proliferarse tan pronto se comenzó a popularizar el uso del Internet gráfico. Los Portales Web tuvieron una especial atención debido a que eran para muchos usuarios su punto de partida cuando querían entrar a Internet. Muchas empresas como American Online, Walt Disney, AT&T y CBS entre otras, se dieron a la tarea de diseñar e implementar Portales Web.

Muchos de los primeros portales solamente contenían directorios Web como Yahoo! y/o motores de búsqueda como Infoseek y Altavista, entre otros. La mayoría de estos portales iniciales se enfocaban en atender usuarios prácticamente en cualquier parte del mundo. Un poco más tarde, se comienza a ver el uso de los portales dentro de las organizaciones como una herramienta que apoya la gerencia del conocimiento al proporcionar a los empleados herramientas y procedimientos que antes eran imposibles de utilizar.

Los Portales Web han estado en constante cambio, no solo por la cantidad de datos que contienen. Poco a poco se ha pasado de modelos basados en páginas estáticas a aplicaciones Web de alta complejidad que gestionan contenidos en múltiples idiomas, integran aplicaciones de colaboración entre los usuarios y proporcionan contenidos en diversos formatos para diferentes dispositivos.

Cuba en aras de lograr una cultura digital en la sociedad, debe aplicar la informatización en todas sus esferas, de ahí la necesidad de dominar e introducir las tecnologías de la información y las comunicaciones. A pesar de ser un país bloqueado económicamente, se apuesta por el desarrollo de la informática para el progreso de la economía y en este sentido se han creado los Politécnicos de Informática, los Joven Club de Computación y Electrónica, se han introducido las carreras de Ciencias Informáticas y de Computación en todas las universidades del país, el surgimiento de la Universidad de las Ciencias Informáticas (UCI) como centro rector de la industria cubana de producción de software en el año 2002 y la creación en el año 2006 de las tres Facultades Regionales de la UCI.

La Universidad de las Ciencias Informáticas se ha venido desarrollando gradualmente en algunas esferas como la docencia, la investigación, la producción, entre otras. Debido al gran volumen y variedad de datos que viajan por la red universitaria, la UCI se mantiene actualizada a través de diferentes sistemas que brindan información como: el Portal Digital de la universidad, el canal informativo y el correo electrónico, entre otros medios. La Universidad está dividida en 10 facultades ubicadas en Ciudad de la Habana y tres facultades regionales ubicadas en La Habana, Granma y Ciego de Ávila respectivamente, cada una de ellas se especializa en una rama distinta según los proyectos que ocupa.

En la actualidad existe un gran avance en el desarrollo de sitios Web en todo el mundo, la Universidad de las Ciencias Informáticas no se ha quedado atrás, pues han surgido nuevas formas de divulgar la información por toda la red. Cada día se suman sitios y páginas Web a la intranet de la universidad así como otros portales Web de importancia.

La Facultad 6 se ha integrado a este proceso de informatización y publicación de la información con la creación del Portal Web de la FEU con que cuenta actualmente, el cual brinda información referente a noticias relevantes, encuestas, foros de discusión y otros temas de interés; pero no muestra información relacionada con la producción, la investigación, la administración de la facultad, la docencia y temas típicos de cada departamento. Pues la facultad no cuenta con un Portal Web que integre todas estas áreas y permita a los estudiantes y trabajadores conocer todos los elementos necesarios o de su interés provenientes de cada área y que la facultad a través de este Portal Web sea identificada en la universidad.

Por lo que se identificó el siguiente **problema científico**: ¿Cómo mantener informada a la comunidad universitaria de la Facultad 6 de la Universidad de las Ciencias Informáticas?

Este problema se enmarca en el **objeto de estudio**: Proceso de gestión de la información.

El **campo de acción** abarcado es: Proceso de gestión de la información en la Facultad 6.

Objetivo general de la investigación: Desarrollar un Portal Web que permita mantener informada de forma sistemática a la comunidad universitaria de la Facultad 6 de la Universidad de las Ciencias Informáticas.

Objetivos específicos de la investigación:

- ✓ Identificar las funcionalidades que debe cumplir el sistema.
- ✓ Realizar análisis y diseño de una interfaz Web adecuada.

- ✓ Diseñar e implementar una base de datos capaz de almacenar la información requerida por las diferentes áreas de la facultad.
- ✓ Implementar los componentes del sistema.

Tareas a desarrollar para asegurar el cumplimiento de los objetivos trazados:

- ✓ Estudio de portales Web existentes en universidades nacionales e internacionales.
- ✓ Análisis del tipo de información que se maneja en las distintas áreas de la facultad 6.
- ✓ Realización del levantamiento de requisitos.
- ✓ Estudio y selección de las herramientas, metodologías y tecnologías posibles a utilizar en el diseño de portales Web.
- ✓ Estudio y selección de la arquitectura a utilizar en el diseño del Portal Web.
- ✓ Estudio de mecanismos o tecnologías adecuados para el acceso a datos.
- ✓ Implementación de la primera versión del producto.

El documento está estructurado de la siguiente forma:

Capítulo1: Fundamentación Teórica. Este capítulo incluye el estado del arte del tema tratado, a nivel nacional e internacional. Se describen las técnicas, tecnologías y metodologías en las que se apoya para la solución del problema.

Capítulo2: Características del Sistema. Se describe el funcionamiento del sistema a través del Modelo de dominio. Se definen además las reglas del negocio, los requisitos funcionales y no funcionales así como los actores y casos de uso del sistema. Se realizan también las descripciones de los casos de uso del sistema.

Capítulo3: Descripción de la Solución Propuesta. En este capítulo se describe como se desarrolla la solución del sistema con la integración de un componente al CMS Drupal, además se describe la arquitectura del sistema, se muestra el diagrama de despliegue, los prototipos de interfaz de usuario, el mapa de navegación y el modelo físico de la base de datos.

Capítulo4: Implementación. En este capítulo se realizan los diagramas de componentes y se muestran fragmentos de código de las clases principales con su descripción correspondiente.

CAPÍTULO 1

FUNDAMENTACIÓN TEÓRICA

Introducción.

Este capítulo es el resultado de la búsqueda y del análisis de la información vinculada al objeto de estudio, se realiza el análisis del estado del arte del tema tratado y se describe la justificación de las herramientas, tecnologías y metodologías utilizadas para la solución del problema planteado.

1.1 Aplicaciones cliente-servidor. Los Portales Web.

1.1.1 Aplicaciones cliente-servidor.

La arquitectura cliente-servidor permite al usuario en una máquina llamada cliente, requerir algún tipo de servicio de una máquina a la que está conectado, llamado servidor, mediante una red LAN (Red de Área Local) o una red WAN (Red de Área Mundial). Estos servicios pueden ser peticiones de datos de una base de datos, de información contenida en archivos o los archivos en sí mismos, o peticiones de imprimir datos en una impresora asociada.

“Este tipo de arquitectura es la base para todas las aplicaciones que hoy se pueden encontrar en Internet y además en la mayoría de las Intranet corporativas que hoy existen en muchas de las empresas.” [1]

Las aplicaciones cliente-servidor son una relación entre procesos que corren en máquinas separadas donde el cliente es el que solicita algún tipo de servicio y el servidor es el que responde las solicitudes.” (Ver Anexo 1)

1.1.2 Internet/Intranet: ¿Cuál es la diferencia?

“El **Internet**, algunas veces llamado simplemente "La Red", es un sistema mundial de redes de computadoras, un conjunto integrado por las diferentes redes de cada país del mundo, por medio del cual un usuario en cualquier computadora puede, en caso de contar con los permisos apropiados, acceder información de otra computadora y poder tener inclusive comunicación directa con otros usuarios en otras computadoras.” [2]

Las **Intranets** son redes privadas, basadas en los estándares y protocolos abiertos desarrollados en Internet. Estos estándares abiertos soportan aplicaciones y servicios como: correo electrónico, acceso a base de datos, servicio de directorio y administración del sistema. Aportan la interfaz de exploración

del World Wide Web (WWW) a la información y servicios almacenados en una red de área local corporativa.

“Las intranets reducen el costo de mantenimiento de una red interna y al mismo tiempo, aumentan la productividad, ya que ofrecen a los usuarios acceso más eficaz a la información y a los servicios que necesitan.” [1]

“Lo que diferencia Internet de una Intranet es la localización de la información y quien accede a ella. Internet es público, global, abierto a cualquiera que tenga una conexión. Las Intranet están restringidas a aquellas personas que están conectadas a la red privada a donde ella pertenece. Ambas funcionan esencialmente de la misma manera, con la misma tecnología TCP/IP para regular el tráfico de datos. Sin embargo desde el punto de vista de las aplicaciones (e-mail, newsgroups y transferencia de archivos FTP, además de la Web), no existe ninguna diferencia entre Internet e Intranet.” [1]

1.1.3 Portales Web.

“Portal es un término, sinónimo de puente, para referirse a un Sitio Web que sirve o pretende servir como un sitio principal de partida para las personas que se conectan al World Wide Web. Los portales tienen gran reconocimiento en Internet por el poder de influencia que tienen sobre grandes comunidades.” [2]

Los portales están dirigidos principalmente a resolver las necesidades específicas de un grupo de personas o de acceso a la información y servicios de una institución pública o privada. Un Sitio Web no alcanza el rango de portal solamente por tratarse de un sitio robusto o por contener información relevante. Un portal es más bien una plataforma de despegue para la navegación en la Web donde se reúne toda la información que los usuarios necesitan en un único lugar para acceder a ella de forma coherente con el objetivo de aumentar la productividad.

Características claves de los portales Web:

- ✓ Se basan en los servicios: Los portales disponen de pequeñas cantidades de funcionalidad que el usuario necesita y que se denominan servicios, un servicio está relacionado con una función particular.
- ✓ Agrupación: Los portales reúnen toda la información y los servicios que los usuarios necesitan en un único sitio y los pone a su disposición mediante una interfaz común.

- ✓ Seguridad: Una vez que el usuario accede al portal, no necesita identificarse en cualquiera de los otros sistemas, ya sea porque utiliza el mismo sistema de identificación o porque el portal puede identificarle en ellos.
- ✓ Neutralidad: Los portales permiten que los usuarios utilicen servicios de la institución.
- ✓ Adaptabilidad: Los portales se basan en roles; esto significa que el portal tiene un conocimiento del rol del usuario y de los servicios que probablemente necesite.

Los portales ofrecen una serie de servicios adicionales como:

- Servicios de búsqueda que incluyen mecanismos de búsqueda, directorios.
- Contenidos, es decir, sitios con información necesaria en áreas de interés, información de varios tópicos como: noticias importantes, deportes, lista de eventos locales, temas docentes, de producción, de investigación.
- Servicios de comunicación como los chat y servicios de mensajería.
- Servicios para el intercambio de ideas como los foros de discusión.

Portales de Universidades Internacionales.

Portal de la Universidad de Oviedo.

“La Web institucional de la Universidad de Oviedo se desarrolló sobre un CMS creado por el administrador del portal. Se trata de ZList que funciona sobre Zope, en la actualidad se encuentra en desarrollo ya que es un sistema exclusivamente a la medida de las necesidades de la universidad. La idea del CMS ZList está inspirada en tres condiciones:

1. Toda página publicada puede ser modificada fácilmente por cualquier persona sin conocimientos técnicos, ni herramientas especiales, con la utilización de una interfaz Web estándar para todas las páginas.
2. La información debe estar almacenada en un único lugar aunque pueda ser publicada en páginas diversas con distintos formatos donde el acceso para modificarla debe estar restringido a sus "dueños".
3. Debe existir un modelo de página por defecto, ya listo para usarse sin intervención de un desarrollador Web, pero ese modelo debe lograr adaptarse con facilidad a cualquier esquema de contenido manteniendo las dos condiciones anteriores.” [3]

<http://www.uniovi.es>

Portal de la Universidad Regiomontana.

“El desarrollo del Portal de la Universidad Regiomontana está basado en tecnología.Net 2.0, para la cual se utilizó un CMS de código abierto llamado DotNetNuke, algunos banners que están en formato swf, así como elementos del skin del portal y contiene algunos componentes en flash.” [4]

<http://www.ur.mx>

Portales de Universidades Nacionales.

Portal de la Facultad de Matemática y Computación de la Universidad de La Habana.

El portal brinda acceso a una serie de informaciones como planes de estudio, cursos de superación, maestrías, la estructura organizativa de la facultad. Ofrece servicios de búsqueda de información, biblioteca, estadísticas de acceso. Para el desarrollo del mismo se utilizaron dos CMS: Rainbow y DotNetNuke versión 3.

“Actualmente se trabaja en una nueva versión del sitio de la facultad, que se montará sobre DotNetNuke 4.05.05 ya que es bastante fácil de instalar y usar, muestra una interfaz Web amena para modificar el contenido de las páginas y se le pueden integrar nuevos controles Web con conocimientos básicos de ASP.NET.” [5]

www.matcom.uh.cu

Portal Digital de la Universidad de las Ciencias Informáticas.

El portal digital de la UCI juega un papel fundamental en el esparcimiento de la información a través de la intranet de la universidad a la que pertenece. Ofrece una serie de informaciones políticas, culturales, vínculos a las diferentes áreas. Brinda servicios de búsqueda como: directorio, guía telefónica, biblioteca, cuenta también con boletines informativos, enlaces a páginas de Cuba, al mapa del centro; toda esta información le permite a los estudiantes y trabajadores interactuar y ampliar sus conocimientos a través de la red universitaria.

<http://intranet.uci.cu>

Portales existentes en la Facultad 6.

Portal de la FEU.

Este portal se encuentra implementado sobre el CMS NukeEt. Cuenta con una sección de noticias, se publican artículos relacionados con eventos que se realizan en la facultad como: Jornada Científica Estudiantil, Mi Web x Cuba, Copa Pascal, Movimiento Juvenil Martiano, entre otros. Presenta vínculos al TSU, la guardia estudiantil, información sobre la FEU. Brinda servicios de mensajería, foro de discusión, información bastante pobre sobre la docencia.

<http://feu6.uci.cu/>

La realización de un Portal Web para la Facultad 6 es muy importante, debido a la necesidad que existe de mantener informada a la facultad y brindar diferentes servicios a los estudiantes y trabajadores, para lograr una comunicación más rápida con las diferentes áreas, intercambiar ideas y experiencias con las restantes facultades, lo que facilita un mejor funcionamiento en la facultad.

1.2 Sistema de Gestión de Contenido (CMS).

Los servicios de información y documentación accesibles a través de Internet más concretamente mediante servidores Web, han aumentado de una forma exponencial. Se han sustituido las páginas y documentos estáticos por documentos generados de forma dinámica. Las herramientas para este trabajo han recibido la denominación de sistemas de gestión de contenidos.

1.2.1 ¿Qué es un CMS?

“Un sistema de gestión de contenido (Content Management System o CMS) es un software que se utiliza principalmente para facilitar la gestión de webs, ya sea en Internet o en una intranet, y por eso también son conocidos como gestores de contenido web (Web Content Management o WCM).” [6]

La funcionalidad de un sistema de gestión de contenidos se puede dividir en varias categorías principales:

- Creación de contenido
- Administración de contenido
- Publicación
- Presentación

La creación de contenido provee una serie de herramientas que les permite a los creadores del sitio sin conocimientos técnicos en sitios Web para que puedan concentrarse en el contenido. Los CMS también permiten administrar la estructura del sitio, donde va cada página y como se enlazan entre sí, brindan la posibilidad de administrar el sitio vía remota, proporcionándole de esa forma mayor capacidad y flexibilidad al sitio.

Los sistemas de administración de contenido tienen poderosos motores que permiten la publicación de la apariencia y el diseño de páginas de la Web que se aplicará automáticamente durante la edición. El CMS para su presentación adopta también una variedad de funcionalidades para elevar la calidad y efectividad del sitio. También puede proporcionar compatibilidad con los diferentes navegadores disponibles en las plataformas: Windows, Linux, Mac, entre otras. Su capacidad de internacionalización lo permite adaptarse al idioma, sistema de medidas y cultura del visitante.

Ventajas de los sistemas de gestión de contenidos:

- Capacidad de manejar eficientemente gran cantidad de páginas Web.
- Trabajan en un ambiente de páginas Web interactivas, es decir, que se generan según las peticiones de los usuarios.
- Controlan el acceso de los usuarios al sistema, mediante su contraseña y permisos asignados a cada uno de ellos.
- Orden en el sistema, ya que se le puede asignar un mismo estilo a todas las páginas generadas.
- Permiten tener una apariencia y navegación uniforme en todo el sitio.
- Aumento de la seguridad.
- Reducción de los gastos de mantenimiento del sitio.

1.2.2 CMS de código abierto y propietarios.

Los CMS según el tipo de licencia escogido se pueden clasificar en CMS de código abierto u open source y CMS propietarios o privados.

Los sistemas de gestión de contenido comercializados por empresas que consideran el código fuente un activo más que tienen que mantener en propiedad y que no permiten que terceros tengan acceso, son los denominados propietarios o privados. Y los de código fuente abierto u open source, son los

desarrollados por individuos, grupos o empresas que permiten el acceso libre y la modificación del código fuente.

Utilizar un CMS de código abierto tiene una ventaja que hace decidirse a la mayoría de usuarios: su costo. Usualmente el software de código abierto es de acceso libre, por lo que la licencia es gratis. Solamente en casos aislados se hacen distinciones entre empresas y entidades sin ánimo de lucro o particulares. En comparación, los productos comerciales pueden llegar a tener un costo que solo una gran empresa puede asumir.

Sin embargo los CMS open source presentan un gran problema y es relacionado con la documentación, ya que por lo general es escasa, dirigida a usuarios técnicos o mal redactada. Este problema se pone de manifiesto cuando los módulos son desarrollados por terceras personas, debido a que estas personas no siempre incorporan las instrucciones de su funcionamiento de forma completa y entendible.

1.2.3 Principales CMS de código abierto.

Post Nuke.

Es un CMS derivado de PHP-Nuke, con licencia open source GNU/GPL. El código de Post Nuke es orientado a objetos y muy modular. Está escrito en PHP y guarda la información en una base de datos MySQL. Las últimas versiones permiten usar cualquier base de datos compatible con el controlador ADOdb. Presenta un sistema de búsqueda integral, soporte multilinguaje con 36 idiomas incluidos y un complejo sistema de permisos para usuarios.

Post Nuke permite incluir en una página Web:

- Lectura de fuentes RSS.
- Chat.
- Encuestas de opinión.
- Traducciones a varios idiomas.
- Galerías de fotos.
- Comercio electrónico.

Desventajas:

- Posee menos módulos iniciales.

- Falta de documentación.
- Sus versiones no presentan una sistemática actualización.

Mambo.

Está descrito en el lenguaje PHP. Posibilita la creación y mantenimiento de sitios Web y portales de manera fácil y dinámica, permitiendo al dueño o administrador de una página Web la simplicidad para actualizarla y hacerla accesible a todo tipo de usuarios a través de una variedad de instrumentos. La simplicidad de Mambo radica en que no son necesarios conocimientos técnicos, ni especializados para crear, mantener, actualizar o personalizar los contenidos de un sitio Web.

Dentro de las principales características de Mambo se encuentran:

- ✓ La creación y administración rápida de una comunidad en línea.
- ✓ Permite gestionar y administrar los usuarios registrados.
- ✓ Permite editar o eliminar opiniones, artículos.
- ✓ Tiene soporte para más de 20 lenguajes.

Desventajas:

- Presenta poca ayuda online.

PHP Nuke.

“El sistema es liberado como software libre bajo la licencia GNU/GPL. Hasta la versión 7.5 se podía descargar gratuitamente desde la Web oficial de PHP-Nuke; la versión 7.5 es la primera que requirió un pago de 10 dólares. La última versión, la 8.0 requiere un pago de 12 dólares para ser bajada desde dicha Web. Esto está permitido según la licencia GNU/GPL (con la condición de que el código fuente también sea entregado), y el comprador tiene a su vez el derecho de distribuir el código fuente gratuitamente. El requisito del pago es aplicable solo a la última versión del programa; cuando una nueva versión sale, la anterior se libera y se puede descargar sin costo.” [1]

Es un sistema basado en tecnologías PHP y MySQL. Soporta más de 20 idiomas, incluyendo inglés y español. PHP-Nuke se ha convertido en una herramienta sofisticada y robusta. PHP-Nuke es ideal para construir un sitio Web comunitario, donde los visitantes pueden crear cuentas, comentar e interactuar con el sitio y contribuir con material de forma sencilla.

Desventajas:

- Los temas presentan el mismo diseño.
- Vulnerabilidad y compatibilidad con algunos navegadores.
- Es necesario nociones básicas sobre la web.
- Consume muchos recursos, lo que hace que el servidor se comporte más lento, afectando incluso a otros sitios hospedados en el mismo servidor.

Joomla.

Joomla es un sistema gestor de contenidos dinámicos (CMS o Content Management System) que permite crear sitios Web de alta interactividad, profesionalidad y eficiencia. Está programado en lenguaje PHP y SQL.

Con este CMS podemos crear sitios Web de noticias, sitios corporativos, sitios Web de presencia, portales comunitarios, e incluso también pueden crearse sistemas que funcionen en redes cerradas (Intranets) para gestionar información interna de compañías o empresas de negocios. Presenta una interfaz sencilla y amigable, cualquier persona puede administrar sus propios contenidos Web sin la necesidad de poseer conocimientos técnicos, sin saber lenguaje HTML y sin recurrir a un webmaster cada vez que hay que actualizar alguna información en un sitio Web.

Desventajas:

- Deficiente planificación de la interfaz administrativa.
- Limitación en las opciones para personalizar rangos de usuarios.
- Dependencia excesiva del Javascript en su panel de administración.
- Es un poco lento.

Drupal.

Drupal es un sistema de gestión de contenido muy configurable. Es open source, con licencia GNU/GPL, escrito en PHP, desarrollado y mantenido por una activa comunidad de usuarios. Ofrece una gran consistencia de todo el sistema. Gracias a su flexibilidad y adaptabilidad, así como la gran cantidad de módulos adicionales de los que dispone, hace que sea adecuado para la realización de muchos sitios Web.

Es un sistema robusto de ayuda online y páginas de ayuda tanto para usuarios como para administradores. Todo el contenido es totalmente indexado en tiempo real y se puede consultar en cualquier momento. Muchos de sus módulos proporcionan funcionalidades como: autenticación mediante jabber, mensajes privados, entre otros.

Para la gestión de los usuarios, el CMS permite que los mismos se puedan registrar e iniciar sesión de forma local o mediante un sistema de autenticación externo como: el jabber, blogger u otro sitio Drupal. Para su uso en una intranet, se puede integrar con un servidor LDAP.

Otra característica fundamental, es la posibilidad de establecer permisos basados en roles donde los administradores del CMS no tienen que establecer permisos para cada usuario. En lugar de eso, pueden asignar permisos a un 'rol' y agrupar los usuarios por roles.

Para la gestión de contenidos presenta un sistema de control de versiones que permite seguir y controlar totalmente las actualizaciones periódicas del contenido: qué se ha cambiado, la hora y la fecha, quién lo ha cambiado. El sistema de temas de Drupal separa el contenido de la presentación, lo que permite controlar o cambiar fácilmente el aspecto del sitio Web. Se pueden crear plantillas con HTML y/o con PHP.

Este CMS 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. Este CMS al estar implementado en PHP, es totalmente portable.

Incluye un módulo que permite a los administradores y/o usuarios crear encuestas on-line totalmente configurables, incorpora foros de discusión.

“Actualmente es muy usado en la UCI en el desarrollo de los sitios Web de las Comunidades de Investigación y Desarrollo, donde el 100% de ellos se han realizado sobre Drupal, así como para la realización del Portal de las Misiones de la República Bolivariana de Venezuela como parte de los proyectos de exportación que en ella se realizan, para el que se eligió Drupal, lo que avala todas las características antes expuestas.” [1]

Debido a la variedad de funcionalidades, las ventajas que proporciona a los desarrolladores y las características que brinda el desarrollo del CMS Drupal, se determina usarlo para la realización del Portal Web de la Facultad 6.

1.3 Metodología utilizada, tendencia y tecnologías actuales.

1.3.1 Tecnología del lado del cliente.

JavaScript.

Es un lenguaje de script que es soportado tanto por el navegador Internet Explorer como el Netscape Navigator, lo que le confiere la característica de ser multiplataforma. Es seguro, fiable, fácil de aprender, rápido y potente. Es un lenguaje completamente abierto, lo que permite adecuar en nuestras páginas las funciones que se hicieron en otras páginas. Cuenta con efectos especiales sobre páginas Web, para crear contenidos dinámicos y elementos de la página que tengan movimiento, cambien de color o cualquier otro dinamismo y permite ejecutar instrucciones como respuesta a las acciones del usuario, con lo que se pueden crear páginas interactivas con programas como: calculadoras, agendas, tablas de cálculo, calendarios y validar informaciones entradas por el usuario para verificar su veracidad o su forma de escribirse.

“Javascript es un lenguaje con muchas posibilidades, permite la programación de pequeños scripts, pero también de programas más grandes, orientados a objetos, con funciones, estructuras de datos complejas, etc. Además, Javascript pone a disposición del programador todos los elementos, para que este pueda acceder a ellos y modificarlos dinámicamente.” [7]

HTML.

HTML es el Lenguaje de Marcas Hipertextuales (HyperText Markup Language), diseñado para estructurar textos y presentarlos en forma de hipertexto, que es el formato estándar de las páginas Web. Gracias a los navegadores como: Internet Explorer, Opera, Firefox, Netscape y Safari, el HTML se ha convertido en uno de los formatos más populares y fáciles de aprender que existen para la elaboración de documentos para la Web. Este lenguaje permite la introducción de referencias a otras páginas por medio de los enlaces hipertexto.

1.3.2 Tecnología del lado del Servidor. PHP.

“PHP es el acrónimo de Hypertext Preprocessor es un lenguaje de "código abierto" interpretado, de alto nivel, embebido en páginas HTML y ejecutado en el servidor.” [8]

“Un lenguaje del lado del servidor es aquel que se ejecuta en el servidor Web, justo antes de que se envíe la página a través de Internet al cliente. Las páginas que se ejecutan en el servidor pueden realizar accesos a bases de datos, conexiones en red, y otras tareas para crear la página final que verá el cliente. El cliente solamente recibe una página con el código HTML resultante de la ejecución de la PHP. Como la página resultante contiene únicamente código HTML, es compatible con todos los navegadores.” [9]

Ventajas adicionales de PHP:

- Puede interactuar con muchos motores de bases de datos tales como: MySQL, MS SQL, Oracle, Informix, PostgreSQL, entre otros. Adicionalmente, PHP soporta ODBC (Estándar Abierto de Conexión con Bases de Datos), así que puede conectarse a cualquier base de datos que soporte tal estándar.
- Es independiente de plataforma.
- En cuanto a la seguridad, PHP posibilita configurar el servidor de modo que se permita o rechacen diferentes usos, lo que puede hacer al lenguaje más o menos seguro en dependencia de las necesidades de cada cual.
- Gracias a la gran librería de funciones que posee, está preparado para realizar muchos tipos de aplicaciones Web.
- PHP es open source, lo cual significa que no estás forzado a pagar actualizaciones anuales para tener una versión que funcione.
- Puede ser utilizado en cualquiera de los sistemas operativos como: Linux, muchas variantes Unix, Microsoft Windows, Mac OS X, RISC OS, Rapidez, entre otros.
- PHP generalmente es utilizado como módulo de Apache, lo que lo hace extremadamente veloz.

1.3.3 Sistema de Gestión de Base de Datos.

PostgreSQL.

PostgreSQL intenta ser un sistema de base de datos de mayor nivel que MySQL, a la altura de Oracle, Sybase o Interbase. Opera bajo la licencia BSD. Utiliza el lenguaje SQL92/SQL99. Fue el pionero en muchos de los conceptos existentes en el sistema objeto-relacional actual, incluido más tarde en otros sistemas de gestión comerciales. PostgreSQL es un sistema objeto-relacional, ya que incluye características de la orientación a objetos, como puede ser la herencia, tipos de datos, funciones, restricciones, disparadores, reglas e integridad transaccional. A pesar de estas características, PostgreSQL no es un sistema de gestión de base de datos puramente orientado a objetos.

Principales características de PostgreSQL.

- ✓ Incorpora una estructura de datos array.
- ✓ Permite la declaración de funciones propias, así como la definición de disparadores.
- ✓ Soporta el uso de índices, reglas y vistas.

- ✓ Permite la gestión de diferentes usuarios, como también los permisos asignados a cada uno de ellos.
- ✓ Tiene mejor soporte para triggers y procedimientos en el servidor.

¿Qué elemento falta?

PostgreSQL es un magnífico gestor de base de datos, capaz de competir con muchos gestores comerciales, aunque consume más recursos y carga más el sistema. Es más lento que MySQL y posee menos funciones en PHP.

MySQL.

MySQL es un sistema de gestión de base de datos relacional. Se ofrece bajo la GNU/GPL para cualquier uso compatible con esta licencia. Su diseño multihilo le permite soportar una gran carga de forma muy eficiente, es un gestor muy usado debido a su gran rapidez y facilidad de uso. Esta gran aceptación es debida, en parte, a que existen muchas 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:

- ✓ Consume muy pocos recursos, tanto de CPU como de memoria.
- ✓ Soporta gran cantidad de tipos de datos para las columnas.
- ✓ Dispone de API's en gran cantidad de lenguajes (C, C++, Java, PHP).
- ✓ Gran portabilidad entre sistemas.
- ✓ Soporta hasta 32 índices por tabla.
- ✓ Gestión de usuarios y passwords, manteniendo un muy buen nivel de seguridad en los datos.
- ✓ MySQL funciona sobre múltiples plataformas como: GNU/Linux, Mac OS X, NetBSD, Solaris, Windows 95, Windows 98, Windows NT, Windows 2000, Windows XP, Windows Vista y otras versiones de Windows.

Ventajas:

- ✓ Posee mayor rendimiento.
- ✓ Mejores utilidades de administración (backup, recuperación de errores).

- ✓ Mejor integración con PHP.
- ✓ No presenta límites en el tamaño de los registros.
- ✓ Mejor control de acceso, en el sentido de qué usuarios tienen acceso a qué tablas y con qué permisos.

¿Por qué utilizar MySQL?

MySQL proporciona gran rapidez para los usuarios, ya que fue concebido con esta idea, además por las facilidades y ventajas expuestas anteriormente y por la fácil integración con el CMS Drupal y el lenguaje de programación PHP.

1.3.4 Servidor Web. Apache.

Apache es un servidor Web, flexible, rápido y eficiente, de código fuente abierto, continuamente actualizado y adaptado a los nuevos protocolos (HTTP 1.1). Está disponible para diferentes plataformas como: FreeBSD, NetBSD, OpenBSD, GNU/Linux, Mac OS y Mac OS X Server, Netware, Solaris, Windows, entre otras. Con los diferentes módulos de apoyo que proporciona y con la API de programación de módulos, puede ser adaptado a diferentes entornos y necesidades. Gracias a ser modular se han desarrollado diversas extensiones entre las que destaca PHP, un lenguaje de programación del lado del servidor.

1.4 Metodología de desarrollo.

1.4.1 Metodología Extreme Programming (XP).

Es una de las metodologías de desarrollo de software más exitosa en la actualidad, consiste en una programación rápida o extrema, cuya particularidad es tener como parte del equipo, al usuario final, pues es uno de los requisitos para llegar al éxito del proyecto.

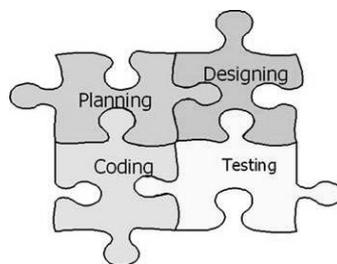


Figura 1. Representación de la Metodología Extreme Programming.

Características de XP, la metodología se basa en:

- **Pruebas Unitarias:** se basa en las pruebas realizadas a los principales procesos, de tal manera que adelantándonos en algo hacia el futuro, podamos hacer pruebas de las fallas que pudieran ocurrir. Es como si nos adelantáramos a obtener los posibles errores.
- **Refabricación:** se basa en la reutilización de código, para lo cual se crean patrones o modelos estándares, siendo más flexible al cambio.
- **Programación en pares:** consiste en que dos desarrolladores participen en un proyecto en una misma estación de trabajo. Cada miembro lleva a cabo la acción que el otro no está haciendo en ese momento. [10]

Lo fundamental en este tipo de metodología es la comunicación entre los usuarios y los desarrolladores, la simplicidad al desarrollar y codificar los módulos del sistema y la retroalimentación concreta y frecuente.

1.4.2 Microsoft Solution Framework (MSF).

Es una metodología flexible e interrelacionada con una serie de conceptos, modelos y prácticas de uso, que controlan la planificación, el desarrollo y la gestión de proyectos tecnológicos. MSF se centra en los modelos de proceso: Modelo de Arquitectura del Proyecto, Modelo de Equipo, Modelo de Proceso, Modelo de Gestión del Riesgo, Modelo de Diseño de Proceso y finalmente Modelo de Aplicación que deja en un segundo plano las elecciones tecnológicas.



Figura 2. Representación de la Metodología MSF.

MSF tiene las siguientes características:

- **Adaptable:** es usado en cualquier parte como un mapa, del cual su uso es limitado a un específico lugar.
- **Escalable:** puede organizar equipos tan pequeños entre 3 o 4 personas, así como también, proyectos que requieren 50 personas o más.
- **Flexible:** es utilizada en el ambiente de desarrollo de cualquier cliente.

- **Tecnología Agnóstica:** porque puede ser usada para desarrollar soluciones basadas sobre cualquier tecnología. [10]

Desventajas de MSF:

- Presenta poca tolerancia al cambio.
- La comunicación está basada en la documentación.
- Se comunica con el cliente solamente al inicio del proyecto.
- Es una metodología impuesta por la empresa.

1.4.3 Rational Unified Process (RUP).

“El Proceso Unificado es un proceso de desarrollo de software. Un proceso de desarrollo de software es el conjunto de actividades necesarias para transformar los requisitos de un usuario en un sistema de software. Sin embargo, el Proceso Unificado es más que un simple proceso; es un marco de trabajo genérico que puede especializarse para una gran variedad de sistemas software, para diferentes áreas de aplicación, diferentes tipos de organizaciones, diferentes niveles de aptitud y diferentes tamaños de proyectos.” [11]

RUP junto con el Lenguaje Unificado de Modelado UML constituye la metodología estándar más utilizada para el análisis, implementación y documentación de sistemas orientados a objetos.

Los aspectos definitorios del RUP se resumen en tres fases fundamentales: dirigido por casos de uso, ya que el desarrollo está dirigido a satisfacer las necesidades de los usuarios del sistema expresadas en casos de uso; centrado en la arquitectura porque incluye los aspectos estáticos y dinámicos más significativos del sistema; iterativo e incremental pues el trabajo está dividido en partes más pequeñas o miniproyectos, donde cada miniproyecto es una iteración que finalmente resulta en un incremento.

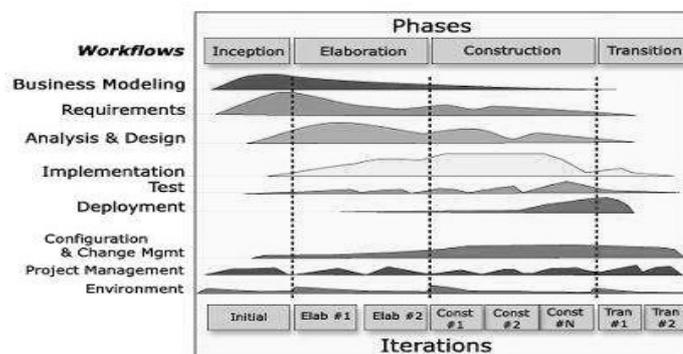


Figura 3. Representación de RUP en Dos Dimensiones.

Flujos de trabajo:

Modelamiento del negocio: Describe los procesos de negocio, mediante la identificación de participantes y actividades que requieren automatización.

Requerimientos: Define qué es lo que el sistema debe hacer, para lo cual se identifican las funcionalidades requeridas y las restricciones que se imponen.

Análisis y diseño: Describe cómo el sistema será realizado a partir de la funcionalidad prevista y las restricciones impuestas (requerimientos).

Implementación: Define cómo se organizan las clases y objetos en componentes, cuáles nodos se utilizarán y la ubicación en ellos de los componentes y la estructura de capas de la aplicación.

Prueba (Testeo): Busca los defectos a lo largo del ciclo de vida.

Instalación: Produce release del producto y realiza actividades (empaquete, instalación, asistencia a usuarios, etc.).

Administración del proyecto: Involucra actividades con las que se busca producir un producto que satisfaga las necesidades de los clientes.

Administración de configuración y cambios: Describe cómo controlar los elementos producidos por todos los integrantes del equipo de proyecto en cuanto a: utilización/actualización concurrente de elementos, control de versiones, etc.

Ambiente: Contiene actividades que describen los procesos y herramientas que soportarán el equipo de trabajo del proyecto; así como el procedimiento para implementar el proceso en una organización.

[12]

Define las siguientes fases:

Conceptualización (Concepción o Inicio): Se describe el negocio y se delimita el proyecto con la descripción de sus alcances con la identificación de los casos de uso del sistema.

Elaboración: Se define la arquitectura del sistema y se obtiene una aplicación ejecutable que responde a los casos de uso que la comprometen.

Construcción: Se obtiene un producto listo para su utilización que está documentado y tiene un manual de usuario. Se obtiene 1 o varios release del producto que han pasado las pruebas.

Transición: El release ya está listo para su instalación en las condiciones reales. Puede implicar reparación de errores. [12]

1.4.3.1 Flujos de Trabajo Definidos.

Dentro de los flujos de trabajo definidos por RUP existen diferentes roles y artefactos de los cuales se implementarán en este trabajo:

FT Modelamiento del Negocio:

Roles:

- Analista de Procesos del Negocio

Artefactos:

- Reglas del Negocio
- Diagrama de Clases del Modelo de Dominio
- Glosario de Términos

FT Requerimientos:

Roles:

- Analista del Sistema
- Especificador de Requerimientos

Artefactos:

- Especificación de los Requerimientos de Software
- Actores del Sistema
- Casos de Uso del Sistema
- Paquete de Casos de Uso
- Modelo de Casos de Uso del Sistema
- Realización de Casos de Uso del Sistema
- Vista de Casos de Uso.

FT Análisis y Diseño:

Roles:

- Diseñador
- Diseñador de Base de Datos
- Diseñador de Interfaz de Usuario

Artefactos:

- Modelo de Despliegue
- Modelo de Datos
- Prototipo de Interfaz de Usuario
- Mapa de Navegación

FT Implementación:

Roles:

- Arquitecto de Software
- Implementador

Artefactos:

- Modelo de implementación
- Componentes

¿Por qué usar RUP?

Por las características explicadas anteriormente, las fases de desarrollo que propone, el trabajo por roles y el cumplimiento de hitos en cada una de las fases garantiza que se desarrolle el proyecto de manera efectiva y eficiente. Además en la facultad existe mayor experiencia en el uso de esta metodología con respecto a otras.

1.5 Lenguaje de Modelado.

El Lenguaje Unificado de Modelado (UML), es un lenguaje de modelado visual que se emplea para especificar, visualizar, construir y documentar artefactos de un sistema de software. Este lenguaje no define un proceso de desarrollo específico, tan solo se trata de una notación. UML es una parte esencial del Proceso Unificado. RUP emplea UML como base para el desarrollo de software en cada una de sus fases y disciplinas. La especificación de UML no define un proceso estándar pero está pensado para ser útil en un proceso de desarrollo iterativo.

UML no es un lenguaje de programación sino un lenguaje de propósito general para el modelado orientado a objetos y también puede considerarse como un lenguaje de modelado visual que permite una abstracción del sistema y sus componentes.

1.6 Herramientas a utilizar.

1.6.1 Herramientas CASE.

Rational Rose.

Es una herramienta de software propietaria utilizada para el modelado visual mediante UML. Rational Rose es la herramienta CASE desarrollada por los creadores de UML (Booch, Rumbaugh y Jacobson), que 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.

Características de Rational Rose:

- ✓ Permite especificar, analizar y diseñar el sistema antes de codificarlo.
- ✓ Permite el chequeo de la sintaxis UML.
- ✓ Genera la documentación de forma automática.
- ✓ Genera código a partir de los modelos.
- ✓ Mantiene la consistencia de los modelos del sistema de software.
- ✓ Permite realizar ingeniería inversa.

Visual Paradigm.

Es una potente herramienta CASE empleada para visualizar y diseñar elementos de software, para ello utiliza el lenguaje UML, proporciona a los desarrolladores una plataforma que les permite diseñar un producto con calidad de forma rápida. Facilita la interoperabilidad con otras herramientas CASE como Rational Rose. Se integra con diversos IDE's como: NetBeans (de Sun), Eclipse (de IBM), JDeveloper (de Oracle), JBuilder (de Borland). Está disponible en varias ediciones: Enterprise, Professional, Community, Standard, Modeler y Personal. Genera código y realiza ingeniería inversa para diferentes lenguajes de programación como: Java, C++, CORBA IDL, PHP, XML Schema y ADA. En adición se genera código para C#, Visual Basic.net, Object Definition Lenguaje (ODL), Flash Action Script, Delphi, Perl y Python. Se integra con el Visio para importar imágenes del mismo para realizar los diagramas de despliegue. Además exporta e importa los diagramas en el estándar XML.

Visual Paradigm es la herramienta CASE que se empleará en la modelación de este proyecto por su característica de ser multiplataforma, por las facilidades que brinda y por ser una herramienta que se puede utilizar legalmente.

DBDesigner.

Se utilizará el DBDesigner por ser una herramienta libre que permite diseñar bases de datos para aplicaciones en internet o cualquier base de datos para una aplicación en Java, C++, Visual C#, Visual Basic, Perl o PHP, para el uso en la Web.

Principales características:

- Disponible para Linux y Windows.
- Modo diseño y modo consulta.
- Ingeniería inversa para las bases de datos MySQL, Oracle, MSSQL, entre otras.
- Generación de esquemas definidos por el usuario.
- Sincronización del modelo a la base de datos.
- Soporte de índices.
- Soporte de entidades débiles.
- Capacidad de documentar la base de datos
- Impresión avanzada de modelos.
- Soporta plugins

1.6.2 Herramientas para el desarrollo de aplicaciones Web.

Quanta Plus.

Quanta Plus es un editor HTML basado en el código de bluefish. Es un proyecto open source, bajo la licencia GPL. Disponible para plataformas Linux, bajo el entorno KDE. Soporta una multitud de lenguajes como: HTML, Javascript, CSS, PHP, SQL, XML, ColdFusion, Perl, DTML, Zope y C++, lo que hace que en la actualidad sea uno de los editores más utilizados en el desarrollo de páginas Web en Linux.

Quanta ofrece varias de funcionalidades:

- Coloreado de sintaxis para todos los lenguajes soportados.

- Cajas de diálogos contextuales, donde se puede elegir las etiquetas HTML que se quieren utilizar.
- Excelente navegador de directorios integrado, con el que se accede a los ficheros fácilmente.
- Completo panel de previsualización, para ver el resultado final del trabajo.
- Validador HTML integrado en la propia aplicación.

Kompozer.

Se utilizará también Kompozer como editor Web por ser una herramienta visual libre, fácil de usar, multiplataforma y con soporte para varios idiomas, basada en Nvu, cuenta con licencia GPL. Creado con el fin de ser una alternativa libre a Adobe Dreamweaver.

Características principales:

- ✓ Edición WYSIWYG (lo que ves es lo que obtienes) lo que posibilita la creación de una Web atractiva de forma sencilla a través de interfaz gráfica sin tener conocimientos de HTML. Esta característica es similar a la ofrecida en Dreamweaver o FrontPage.
- ✓ Incorpora código HTML eficiente, que lo hace capaz de trabajar con los más populares buscadores de hoy. Tiene un potente soporte para formularios, tablas y templates.
- ✓ Facilita el manejo de múltiples archivos.
- ✓ Posee interesantes herramientas de depuración y de gestión de estilos CCS.

Kate.

Se utilizará conjuntamente Kate que es un editor de textos para KDE por las características que se presentan a continuación.

- Es software libre, de código abierto y gratuito, bajo la licencia LGPL.
- Usa las librerías gráficas Qt para mostrar los controles de la interfaz.
- Implementa la tecnología KParts que permite colocar a Kate como un componente de edición en cualquier otra aplicación de KDE.
- Resaltado de sintaxis, extensible mediante archivos XML.
- Seguimiento de código para diferentes lenguajes como son: C++, C y otros.
- Permite mantener varios documentos abiertos en una ventana.

- Cuenta con soporte de sesiones.
- Posee un administrador de archivos básico.

1.6.3 Herramientas para el trabajo con la Base de Datos.

MySQL Query Browser.

Esta herramienta es muy útil para el trabajo con la base de datos MySQL. Es un editor de sentencias SQL visual, que además incorpora herramientas para la optimización de las consultas. Cuenta con un editor de tablas y registros, que permite crear nuevas tablas o cambiar las existentes y posibilita cambiar los registros.

Dispone de los siguientes componentes:

- ✓ Editor de sentencias SQL, que permite crear sentencias de forma visual o manual y moverse por el historial de sentencias que se han generado.
- ✓ Visor de resultados, para que se puedan examinar los resultados devueltos por las consultas y comparar los resultados obtenidos entre varias sentencias SQL. Se pueden tener activas varias consultas a la vez y seleccionar la que se quiere visualizar con una pestaña.
- ✓ Navegador de objetos, para manejar las bases de datos, favoritos o historial. Se puede visualizar aquí un esquema de las distintas bases de datos, con sus diferentes tablas y registros, que se utilizarán luego para generar las sentencias visualmente.
- ✓ Visor de información, con referencia acerca de la base de datos MySQL, con acceso a la sintaxis y librería de funciones.

1.6.4 Herramientas para la gestión de requisitos.

OSRMT (Open Source Requirements Management Tool).

Se utilizará para la gestión de requisitos la herramienta OSRMT ya que fue desarrollada bajo la licencia GPL y escrita en lenguaje Java. Permite la descripción avanzada de diversos tipos de requisitos y garantiza la trazabilidad entre todos los documentos relacionados con la ingeniería de requisitos (funcionalidades, requisitos, casos de uso, casos de prueba).

1.6.5 Ambiente de Desarrollo Integrado (IDE).

Eclipse.

“Eclipse es una plataforma de software de código abierto independiente de una plataforma para desarrollar lo que el proyecto llama "Aplicaciones de Cliente Enriquecido", opuesto a las aplicaciones "Cliente-liviano" basadas en navegadores. Esta plataforma, típicamente ha sido usada para desarrollar un entorno integrado de desarrollo. Este IDE es más general para el desarrollo de aplicaciones en Java, en la que se ha especializado, también permite programar en PHP, aunque con muy pocas opciones de revisión y corrección de errores y sin completamiento de código. Es una herramienta que necesita de mucha ayuda del hardware para realizar la compilación del código fuente escrito, además de ser multiplataforma.” [1]

NetBeans.

NetBeans se refiere a una plataforma para el desarrollo de aplicaciones de escritorio que usa el lenguaje Java. La plataforma NetBeans permite que las aplicaciones sean desarrolladas a partir de un conjunto de componentes de software llamados módulos para el desarrollo de aplicaciones Java en una sola descarga, permitiéndole al usuario comenzar a trabajar inmediatamente. Este IDE soporta el desarrollo de todos los tipos de aplicación Java (J2SE, Web, EJB y aplicaciones móviles).

Zend Studio.

Zend Studio es un editor de texto para páginas PHP que proporciona un buen número de ayudas desde la creación y gestión de proyectos hasta la depuración del código. El programa entero está escrito en Java, lo que ha permitido lanzar con relativa facilidad y rapidez versiones del producto para Windows, Linux y MacOS. Consta de dos partes en las que se dividen las funcionalidades de parte del cliente y las del servidor. La parte del cliente contiene una ayuda contextual con todas las librerías de funciones del lenguaje que asiste en todo momento ofreciendo nombres de las funciones y parámetros que deben recibir. La parte del servidor contiene una herramienta de depuración. Gracias a la cual podemos ejecutar páginas y conocer en todo momento el contenido de las variables de la aplicación y las variables del entorno como las cookies, las recibidas por formulario o en la sesión.

PHPDesigner.

Es un IDE para el desarrollo de sitios Web con PHP, permite trabajar con comodidad en otros lenguajes de programación como HTML, XHTML, CSS y SQL. Cuenta con cliente de FTP y navegador de ficheros integrado, posee utilidades de corrección y autocompletado automático, búsqueda

integrada en Google y soporte para proyectos. Soporta los siguientes formatos: PHP, HTML, XHTML, CSS, Java, Perl, JavaScript, VB, C# y SQL. Se integra con navegadores externos (Internet Explorer, Netscape, Firefox y Opera).

¿Por qué usar PHPDesigner?

Este IDE es el más apropiado para el desarrollo del programa, ya que es una herramienta útil y gratis. Pensado para acelerar y optimizar el proceso de edición, eliminación de errores, análisis y publicación de aplicaciones, además de sitios Web proporcionados por PHP y otros lenguajes Web. Esta herramienta permite ahorrar tiempo, de esta manera aumentará y mejorará la productividad ya que podrá hacer mucho más, como por ejemplo acceder a su clase de funciones, variables, etiquetas y propiedades.

1.7 Conclusiones.

Después de realizar un análisis del estado del arte de las metodologías, herramientas y tecnologías necesarias y actuales. Se decide utilizar el CMS Drupal como sistema de gestión de contenido y con él, desarrollar el Portal Web de la Facultad 6 de la Universidad de las Ciencias Informáticas.

Para el desarrollo del portal fue seleccionada la tecnología LAMP (Linux, Apache, MySQL, PHP), Java Script como lenguaje de programación del lado del cliente por las facilidades que brinda, PHPDesigner como IDE, RUP como metodología de desarrollo a utilizar y UML como lenguaje de modelado, Visual Paradigm y DBDesigner como herramientas CASE, además de las herramientas: Quanta Plus, Kompozer y Kate que se definieron para el desarrollo de la aplicación, la herramienta MySQL Query Browser para el trabajo con la base de datos y la herramienta OSRMT para la gestión de requisitos.

CAPÍTULO 2

CARACTERÍSTICAS DEL SISTEMA

Introducción.

En el presente capítulo se realiza un análisis de la propuesta del sistema. Para ello se describen los conceptos más importantes del dominio, las reglas del negocio, se especifican los requisitos funcionales y no funcionales, se identifica mediante el diagrama de casos de uso del sistema las relaciones entre los actores y casos de uso del sistema; así como las descripciones textuales de los casos de uso.

2.1 Modelo de Dominio.

Después de efectuar un análisis profundo del problema en cuestión, se llega a la conclusión de que en la presente investigación no se definen concretamente todos los procesos y existen múltiples responsabilidades, por lo que se decide dar un nuevo enfoque a todo el proceso. Para ello se realiza un modelo de dominio. Este modelo contribuye posteriormente a identificar algunas clases que se utilizarán en el sistema.

Un modelo de dominio captura los tipos más importantes de objetos que existen o los eventos que suceden en el entorno donde estará el sistema. A continuación se muestra el diagrama de clases del modelo de dominio:

2.1.1 Conceptos del Modelo de Dominio.

Decano: Persona que dirige una facultad dentro de una universidad.

Vicedecano: Asesor del Decano en un área específica.

VD Extensión: Asesor del Decano para atender la rama de extensión universitaria.

VD Formación: Asesor del Decano para atender la rama de docencia.

VD Producción e Investigación: Asesor del Decano para atender la rama de producción e investigación.

Jefe de Departamento: Persona que dirige un departamento de una especialidad.

Mural: Lugar donde se encuentran publicadas las noticias, concursos, avisos y otras informaciones. Permite mantener informado y actualizado al centro donde se encuentre.

Noticia: Información relevante del acontecer diario.

Aviso: Anuncios sobre alguna información relevante.

Concurso: Actividad que permite una amplia participación de las personas en un tema específico donde se expresan sus conocimientos.

Convocatoria: Es el escrito mediante el cual se convoca o invita a un grupo de personas a participar en alguna actividad específica.

Especialista: Persona con ciertos grado y conocimientos científicos en un tema relacionado con una rama específica.

Cliente: Empresa o entidad que solicita algún servicio de la universidad o de la facultad específicamente.

Producto Generado: Módulo, componente u otro tipo de producto que se pueda generar para ser usado posteriormente en algún proyecto.

Máquina: Computadora personal.

Tribunal: Grupo de profesores con la autoridad y responsabilidad de evaluar el trabajo de diploma de estudiantes universitarios.

Estudiante: Persona que cursa estudios en un centro docente.

Profesor: Persona que se dedica a la formación de los estudiantes.

Horario-Docente: Planificación de las actividades presenciales del plan de estudio de una carrera.

AA: Estudiante que además de recibir clases, las imparte a otro grupo de estudiantes como parte de un movimiento estudiantil (Alumnos Ayudantes) o también puede ayudar de otras formas al colectivo al cual pertenece.

Activ Metodológica: Actividad encaminada a la preparación de los profesores en su labor docente-educativa.

Tesis: Ejercicio de culminación de estudio en el cual se exponen los resultados científicos alcanzados por el aspirante en su trabajo de investigación.

Proyecto: Actividad desarrollada por una o varias personas cuyo objetivo es la obtención de un producto o aporte investigativo.

Lab-Producción: Lugar donde se realizan los proyectos productivos.

GOE (Guardia Obrera Estudiantil): Guardia que realizan los estudiantes en el centro para el cuidado del mismo.

TSU (Trabajo Socialmente Útil): Labor que realizan los estudiantes diariamente para colaborar con el mantenimiento y embellecimiento del centro.

2.1.2 Reglas del Negocio.

- Un tema de tesis puede tener hasta 3 autores.
- Un estudiante tiene una única tesis.
- Un profesor pertenece a un solo proyecto.
- Un tribunal tiene un solo presidente.
- Un tribunal tiene un solo vocal.
- Un tribunal tiene un solo secretario.
- Una tesis tiene un solo oponente.

2.2 Requisitos Funcionales.

Los requisitos funcionales son aquellos que responden a: ¿qué debe hacer el sistema? y describen las capacidades que el sistema debe cumplir.

El sistema propuesto tiene los siguientes requisitos funcionales:

- R1 Autenticar Usuario
- R2 Visualizar Proyectos
- R3 Visualizar Profesores
- R4 Visualizar Estudiantes
- R5 Visualizar Tesis
- R6 Visualizar Estudiante-Máquina
- R7 Visualizar Estudiante-Actividad
- R8 Visualizar Estudiante-Proyecto
- R9 Visualizar Estudiante-Tesis
- R10 Visualizar Profesor-Proyecto
- R11 Visualizar Profesor-Tesis
- R12 Mostrar Aviso
- R13 Mostrar Evento
- R14 Mostrar Noticia
- R15 Mostrar Encuesta
- R16 Descargar Fichero
- R17 Mostrar Resultado de Búsqueda
- R18 Mostrar Categorías de Búsqueda
- R19 Editar Perfil de Usuario
- R20 Crear un Nuevo Tema en el Foro
- R21 Enviar Respuesta a un Tema del Foro
- R22 Escribir Mensajes
- R23 Eliminar Mensajes
- R24 Guardar Mensajes
- R25 Adicionar Estudiante

R26 Buscar y Visualizar Estudiantes

R27 Modificar Estudiante

R28 Eliminar Estudiante

R29 Adicionar Profesor

R30 Buscar y Visualizar Profesores

R31 Modificar Profesor

R32 Eliminar Profesor

Paquete Administración del Sistema

R33 Añadir Contenedor de Foro

R34 Editar Contenedor de Foro

R35 Eliminar Contenedor de Foro

R36 Añadir Foro de Discusión

R37 Editar Foro de Discusión

R38 Eliminar Foro de Discusión

R39 Buscar Usuario

R40 Añadir Usuario

R41 Editar Usuario

R42 Eliminar Usuario

R43 Añadir Rol

R44 Editar Rol

R45 Eliminar Rol

R46 Asignar Control de Acceso a Roles

R47 Activar Módulo

R48 Desactivar Módulo

R49 Añadir Bloque

R50 Activar Bloque

R51 Editar Bloque

R52 Desactivar Bloque

R53 Añadir Menú

R54 Añadir Opciones de Menú

R55 Activar Menú

R56 Editar Menú

R57 Desactivar Menú

R58 Eliminar Menú

R59 Activar Tema

R60 Editar Tema

R61 Desactivar Tema

Paquete Gestión de Contenido

R62 Crear Aviso

R63 Editar Aviso

R64 Eliminar Aviso

R65 Adicionar Noticia

R66 Editar Noticia

R67 Eliminar Noticia

R68 Adicionar Evento

R69 Editar Evento

R70 Eliminar Evento

Paquete Producción

R71 Adicionar Especialista

R72 Buscar y Visualizar Especialistas

- R73 Modificar Especialista
- R74 Eliminar Especialista
- R75 Adicionar Cliente
- R76 Buscar y Visualizar Clientes
- R77 Modificar Cliente
- R78 Eliminar Cliente
- R79 Adicionar Proyecto
- R80 Buscar y Visualizar Proyectos
- R81 Modificar Proyecto
- R82 Eliminar Proyecto
- R83 Adicionar Producto Generado
- R84 Buscar y Visualizar Productos Generados
- R85 Modificar Producto Generado
- R86 Eliminar Producto Generado
- R87 Asignar Profesor-Proyecto
- R88 Eliminar Profesor-Proyecto
- R89 Asignar Especialista-Proyecto
- R90 Eliminar Especialista-Proyecto
- R91 Asignar Estudiante-Proyecto
- R92 Eliminar Estudiante-Proyecto
- R93 Asignar Proyecto-Cliente
- R94 Eliminar Proyecto-Cliente
- R95 Asignar Producto Generado-Proyecto
- R96 Eliminar Producto Generado-Proyecto

Paquete Tesis

R97 Adicionar Tesis

R98 Buscar y Visualizar Tesis

R99 Modificar Tesis

R100 Eliminar Tesis

R101 Adicionar Tribunal

R102 Buscar y Visualizar Tribunales

R103 Eliminar Tribunal

R104 Asignar Estudiante-Tesis

R105 Eliminar Estudiante-Tesis

R106 Asignar Profesor-Tesis

R107 Eliminar Profesor-Tesis

R108 Asignar Profesor-Tribunal

R109 Eliminar Profesor-Tribunal

R110 Asignar Tesis-Tribunal

R111 Eliminar Tesis-Tribunal

Paquete Extensión Universitaria

R112 Adicionar Actividad

R113 Buscar y Visualizar Actividades

R114 Modificar Actividad

R115 Eliminar Actividad

R116 Asignar Estudiante-Actividad

R117 Eliminar Estudiante-Actividad

Paquete Horario-Máquina de Proyectos

R118 Adicionar Máquina

R119 Buscar y Visualizar Máquinas

R120 Modificar Máquina

R121 Eliminar Máquina

R122 Asignar Estudiante-Máquina

R123 Eliminar Estudiante-Máquina

2.3 Requisitos no Funcionales.

Los requisitos no funcionales son las propiedades o cualidades que el producto debe tener. Debe pensarse en estas propiedades como las características que hacen al producto atractivo, usable, rápido o confiable. En muchos casos los requisitos no funcionales son fundamentales en el éxito del producto. Normalmente están vinculados a requisitos funcionales, es decir una vez se conozca lo que el sistema debe hacer podemos determinar cómo ha de comportarse, qué cualidades debe tener o cuán rápido o grande debe ser.

Para el sistema propuesto se han definido los siguientes requisitos no funcionales:

Apariencia o interfaz externa:

La aplicación debe contar con una interfaz amigable que le permita al usuario interactuar de forma cómoda con la misma, le agilice y facilite el trabajo con el software. Las páginas no tendrán muchas imágenes y poseerán pocos colores, donde predominará el color verde que identifica a la facultad 6 de la Universidad de las Ciencias Informáticas. La página principal tendrá información que le servirá de guía al usuario.

Usabilidad:

La aplicación tendrá un ambiente sencillo y será fácil de manejar para los usuarios, incluso aquellos que no han tenido mucha experiencia en el trabajo con computadoras o con sistemas informáticos.

Disponibilidad:

La aplicación deberá estar disponible las 24 horas del día los 7 días de la semana y garantizar un acceso de forma fácil y rápida para los usuarios.

Soporte:

El software será actualizado por los administradores del sistema de forma sistemática, también será garantizado su mantenimiento.

Rendimiento:

El sistema tratará de garantizar la rapidez de respuesta ante las solicitudes de los usuarios en el tiempo mínimo posible.

Portabilidad:

El sistema podrá ser ejecutado sobre los sistemas operativos Linux y Windows.

Seguridad:

Los usuarios no necesariamente deberán estar autenticados para visualizar algunos de los contenidos que brinda el portal. Sin embargo los administradores y moderadores se comportarán como usuarios, pero para poder además realizar configuraciones de los contenidos o algún cambio en el sistema, tendrán obligatoriamente que estar logueados. Cada usuario logueado podrá realizar las operaciones correspondientes con su rol definido en el sistema.

Software:

Se debe disponer para poder instalar la aplicación:

En el cliente:

Sistema operativo Linux o Windows (98 o superior).

Navegador Web estándar con capacidad de interpretación de Java Script y CSS compatible con Internet Explorer 5.5 o superior, Netscape, Mozilla 1.7 o superior o Firefox 0.9.3 o superior.

En el servidor:

Servidor Web Apache.

PHP5 o superior como lenguaje de programación.

MySQL 5 o superior como servidor de base de datos.

Hardware

Las PC servidoras deberán contar con un microprocesador Intel Pentium III o superior, a 450MHz o superior, con un mínimo de 512MB de memoria RAM, un disco duro con capacidad de 1GB o superior y una tarjeta de red alámbrica, inalámbrica o módem. En dependencia del volumen de información que posean los archivos adjuntos, se necesitará de mayor capacidad de disco duro.

Legales

Se usarán herramientas de software libre bajo las licencias GNU/GPL.

Restricciones en el diseño y la implementación

El diseño de la aplicación será basado en la Metodología RUP con el uso del lenguaje de modelado UML. Se empleará como herramienta CASE Visual Paradigm para el modelado del sistema y DBDesigner para el modelado de la base de datos. Se usará la tecnología Lamp. Se utilizarán otras herramientas como Quanta Plus, Kompozer, Kate, MySQL Query Browser y OSRMT.

2.4 Descripción del Sistema Propuesto.

El portal debe permitir el acceso de los usuarios o personas a los diferentes bloques y servicios en dependencia del rol que desempeñen dentro de este.

Si la persona que accede al sistema es un usuario que nunca ha sido registrado, podrá visualizar algunos contenidos del portal.

La persona o usuario que decida registrarse, una vez ingresado en el sistema, podrá hacer lo mismo que el usuario que no se registra, además puede observar los usuarios que se encuentran en línea, editar su perfil personal, enviar mensajes privados a cualquier usuario del sistema, participar en el foro discusión, realizar búsquedas y visualizar datos generales.

El usuario registrado que una vez autenticado se compruebe que es personal autorizado de la facultad con los privilegios concedidos por el administrador, podrá gestionar ciertos módulos del sistema.

El sistema contará con un administrador que es la persona que tiene todos los privilegios en la aplicación. Podrá gestionar su configuración, controlar toda la aplicación y el mantenimiento de la misma.

2.5 Definición de los Actores del Sistema.

Los actores representan terceros fuera del sistema que interactúan con él. Cada actor juega un rol determinado al interactuar con el sistema y diferentes usuarios pueden asumir el mismo rol de un actor.

Tabla 1. Descripción de los Actores del Sistema.

Actor	Descripción
Administrador	Representa al editor general del sistema, quien además gestiona la configuración del mismo y tiene todos los privilegios en la aplicación.
Moderador	Representa a un usuario con privilegios concedidos por el administrador para gestionar ciertos módulos del sistema.
Usuario Anónimo	Representa a los usuarios que acceden al sistema sin registrarse en el mismo, actúan como invitados a la aplicación.
Usuario Autenticado	Representa a los usuarios que ingresan en el sistema.

2.6 Definición de los Casos de Uso del Sistema.

“Los casos de uso son fragmentos de funcionalidad que el sistema ofrece para aportar un resultado de valor para sus actores.” [13]

2.6.1 Patrones de Casos de Uso.

“La experiencia en la utilización de casos de uso ha evolucionado en un conjunto de patrones que permiten con más precisión reflejar los requisitos reales, haciendo más fácil el trabajo con los sistemas, y mucho más simple su mantenimiento. Dado un contexto y un problema a resolver, estas técnicas han mostrado ser la solución adoptada en la comunidad del desarrollo de software. Se presentan a modo de herramientas que permiten resolver los problemas que se les planteen a los desarrolladores de una forma ágil y sistemática. Estos patrones se enfocan hacia el diseño y las técnicas utilizadas en modelos de alta calidad, y no en cómo modelar usos específicos.” [14]

Como patrón utilizado se encuentra:

CRUD (Creating, Reading, Updating, Deleting).

"Este patrón consta de un caso de uso, llamado Información CRUD o Gestionar información, modela todas las operaciones que pueden ser realizadas sobre una parte de la información de un tipo específico, tales como creación, lectura, actualización y eliminación. Suele ser utilizado cuando todos los flujos contribuyen al mismo valor del negocio, y estos a su vez son cortos y simples." [14]

2.6.2 Listado de Casos de Uso.

CU-1 Autenticar Usuario

CU-2 Visualizar Datos Generales

CU-3 Visualizar Contenido

CU-4 Realizar Búsqueda

CU-5 Editar Perfil de Usuario

CU-6 Enviar Tema al Foro

CU-7 Gestionar Mensajes Privados

CU-8 Adicionar Estudiante

CU-9 Editar Estudiante

CU-10 Adicionar Profesor

CU-11 Editar Profesor

Paquete Administración del Sistema

CU-12 Administrar Foro de Discusión

CU-13 Gestionar Usuario

CU-14 Gestionar Rol

CU-15 Activar Módulo

CU-16 Gestionar Bloque

CU-17 Gestionar Menú

CU-18 Gestionar Temas

Paquete Gestión de Contenido

CU-19 Gestionar Avisos

CU-20 Gestionar Noticias

CU-21 Gestionar Eventos

Paquete Producción

CU-22 Adicionar Especialista

CU-23 Editar Especialista

CU-24 Adicionar Cliente

CU-25 Editar Cliente

CU-26 Adicionar Proyecto

CU-27 Editar Proyecto

CU-28 Adicionar Producto Generado

CU-29 Editar Producto Generado

CU-30 Asignar Profesor-Proyecto

CU-31 Eliminar Profesor-Proyecto

CU-32 Asignar Especialista-Proyecto

CU-33 Eliminar Especialista-Proyecto

CU-34 Asignar Estudiante-Proyecto

CU-35 Eliminar Estudiante-Proyecto

CU-36 Asignar Proyecto-Cliente

CU-37 Eliminar Proyecto-Cliente

CU-38 Asignar Producto Generado-Proyecto

CU-39 Eliminar Producto Generado-Proyecto

Paquete Tesis

CU-40 Adicionar Tesis

CU-41 Editar Tesis

CU-42 Adicionar Tribunal

CU-43 Editar Tribunal

CU-44 Asignar Estudiante-Tesis

CU-45 Eliminar Estudiante-Tesis

CU-46 Asignar Profesor-Tesis

CU-47 Eliminar Profesor-Tesis

CU-48 Asignar Profesor-Tribunal

CU-49 Eliminar Profesor-Tribunal

CU-50 Asignar Tesis-Tribunal

CU-51 Eliminar Tesis-Tribunal

Paquete Extensión Universitaria

CU-52 Adicionar Actividad

CU-53 Editar Actividad

CU-54 Asignar Estudiante-Actividad

CU-55 Eliminar Estudiante-Actividad

Paquete Horario-Máquina de Proyectos

CU-56 Adicionar Máquina

CU-57 Editar Máquina

CU-58 Asignar Estudiante-Máquina

CU-59 Eliminar Estudiante-Máquina

2.7 Diagrama de Casos de Uso del Sistema.

El diagrama de casos de uso del sistema ayuda a comprender gráficamente los procesos del sistema y su interacción con los actores.

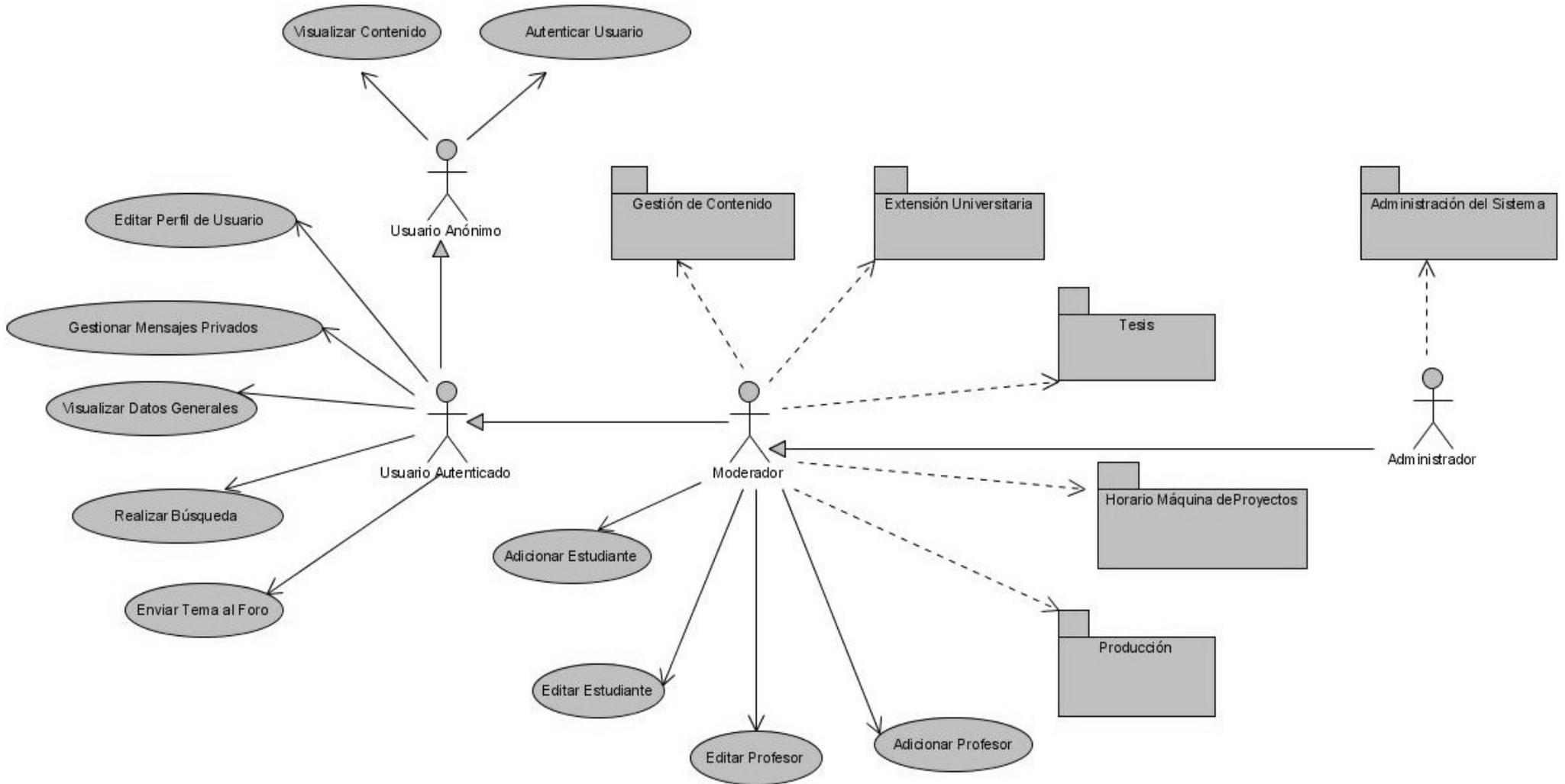


Figura 5. Diagrama de CUS General.

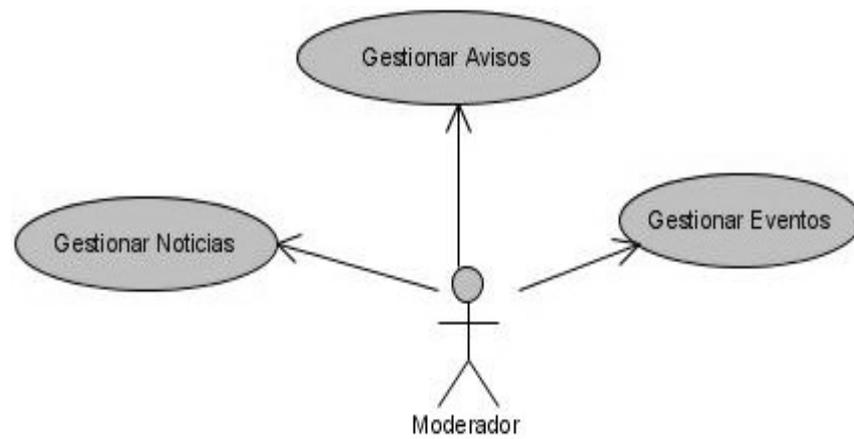


Figura 6. Diagrama de CUS correspondiente al paquete Gestión de Contenido.

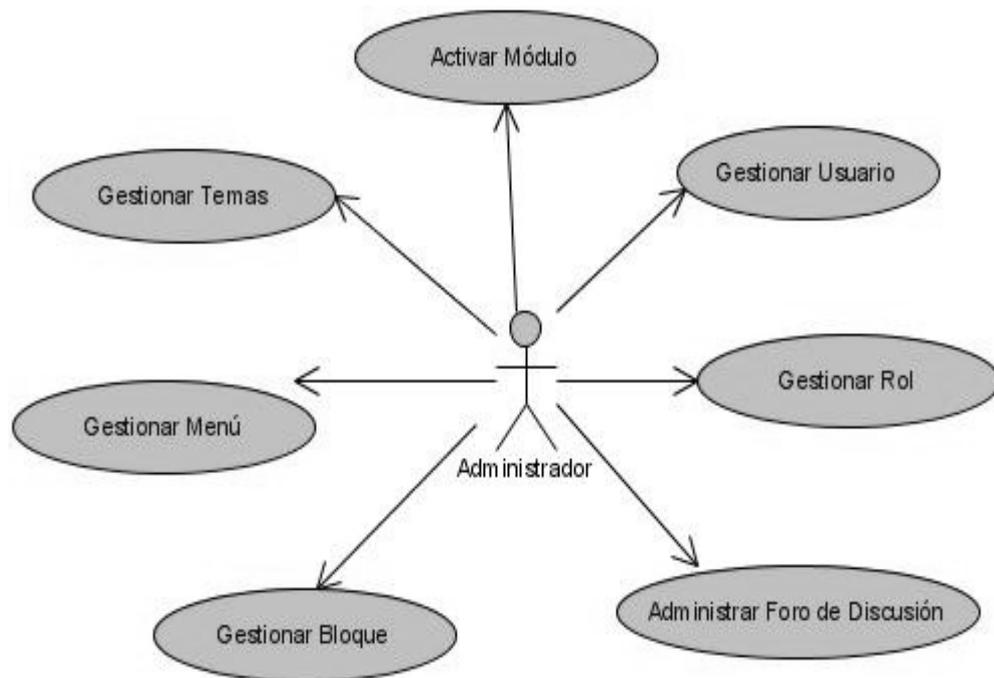


Figura 7. Diagrama de CUS correspondiente al paquete Administración del Sistema.

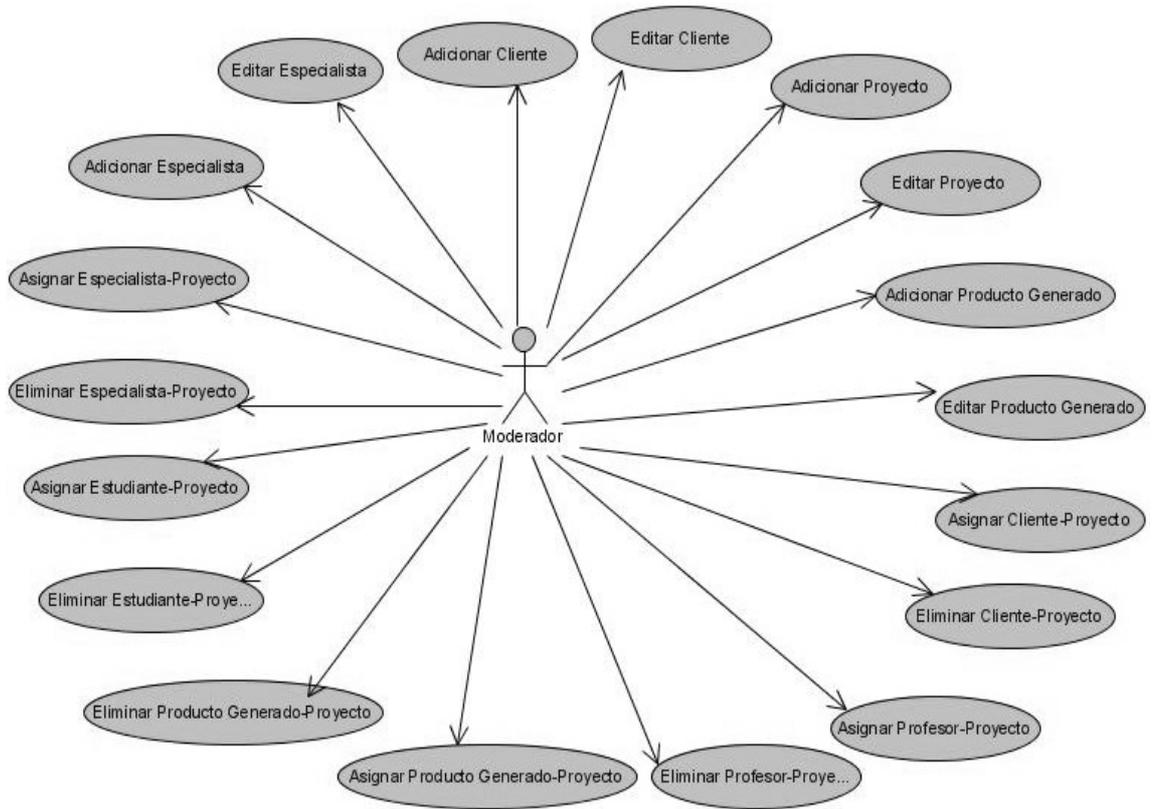


Figura 8. Diagrama de CUS correspondiente al paquete Producción.

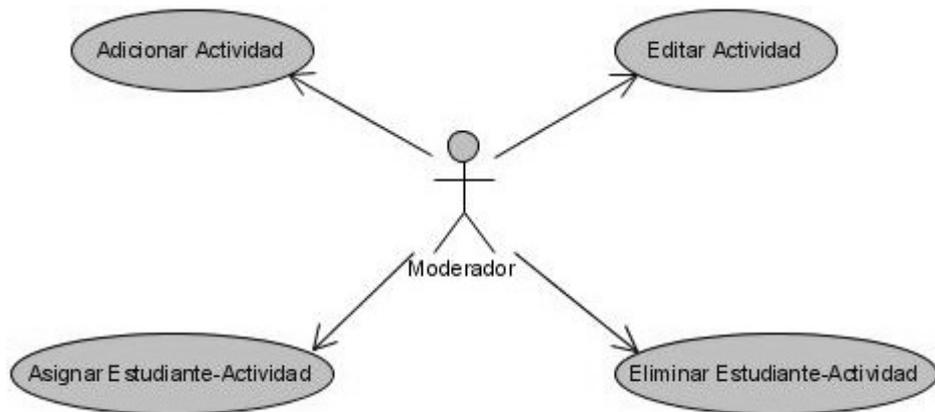


Figura 9. Diagrama de CUS correspondiente al paquete Extensión Universitaria.

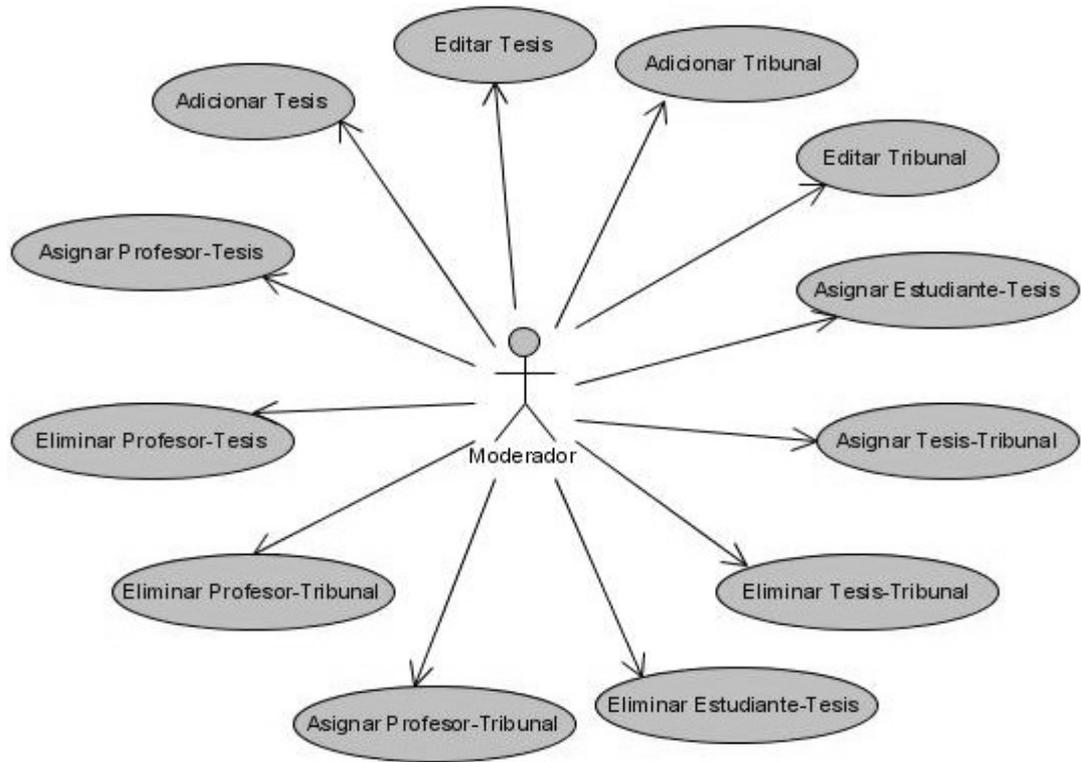


Figura 10. Diagrama de CUS correspondiente al paquete Tesis.

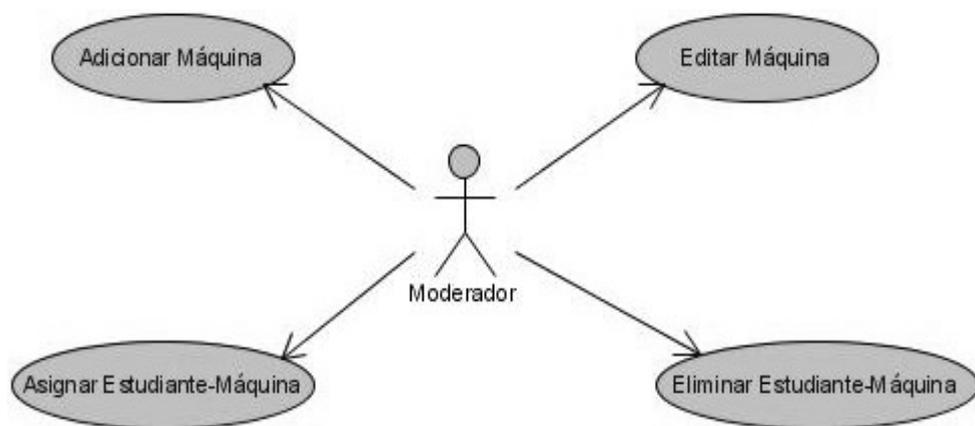


Figura 11. Diagrama de CUS correspondiente al paquete Horario-Máquina de Proyectos.

2.8 Descripción de los Casos de Uso del Sistema.

Mediante la descripción de los casos de uso del sistema se detalla la secuencia de eventos que los actores llevan a cabo para completar un proceso a través de la aplicación.

A continuación se muestra la descripción resumida del caso de uso: Visualizar Datos Generales y de algunos casos de uso correspondientes al paquete Producción (Adicionar Proyecto, Editar Proyecto, Asignar Profesor-Proyecto, Eliminar Profesor-Proyecto, Asignar Estudiante-Proyecto, Eliminar Estudiante-Proyecto), las descripciones restantes no se muestran en el documento pues tienen un comportamiento similar. Las descripciones ampliadas se encuentran en el expediente de proyecto.

Tabla 2. Descripción del caso de uso Visualizar Datos Generales.

Caso de Uso	
CU-2	Visualizar Datos Generales
Actores	Usuario Autenticado (inicia).
Resumen	<p>El caso de uso inicia cuando el Usuario Autenticado selecciona una de las siguientes opciones:</p> <ul style="list-style-type: none"> • Visualizar Proyectos • Visualizar Profesores • Visualizar Estudiantes • Visualizar Tesis • Visualizar Estudiante-Máquina • Visualizar Estudiante-Actividad • Visualizar Estudiante-Proyecto • Visualizar Estudiante-Tesis • Visualizar Profesor-Proyecto • Visualizar Profesor-Tesis <p>El sistema le muestra la interfaz correspondiente según la opción</p>

	seleccionada y ejecuta las acciones necesarias. El caso de uso finaliza cuando se emite el resultado de la opción solicitada.
Referencias	R2, R3, R4, R5, R6, R7, R8, R9, R10, R11
Precondiciones	El Usuario Autenticado se encuentra logueado.
Poscondiciones	El sistema busca y visualiza datos generales de la facultad.

Tabla 3. Descripción del caso de uso Adicionar Proyecto.

Caso de Uso	
CU-26	Adicionar Proyecto.
Actores	Moderador (inicia).
Resumen	El caso de uso inicia cuando el Moderador escoge Adicionar Proyecto. El sistema muestra la interfaz para adicionar datos del proyecto. El caso de uso finaliza cuando se emite el resultado de la opción solicitada.
Referencias	R79
Precondiciones	El Moderador se encuentra autenticado.
Poscondiciones	Es adicionado un nuevo proyecto.

Tabla 4. Descripción del caso de uso Editar Proyecto.

Caso de Uso	
CU-27	Editar Proyecto.
Actores	Moderador (inicia).
Resumen	El caso de uso inicia cuando el Moderador escoge Editar Proyecto. El sistema muestra la interfaz para editar datos de los proyectos, dando la posibilidad de buscar para eliminar y/o modificar. El caso de uso finaliza

	cuando se emite el resultado de la opción solicitada.
Referencias	R80, R81, R82
Precondiciones	El Moderador se encuentra autenticado.
Poscondiciones	Los datos de los proyectos son editados (modificados o eliminados).

Tabla 5. Descripción del caso de uso Asignar Profesor-Proyecto. (Descripción ampliada: Ver Anexo 2)

Caso de Uso	
CU-30	Asignar Profesor-Proyecto
Actores	Moderador (inicia).
Resumen	El caso de uso inicia cuando el Moderador escoge Asignar Profesor-Proyecto. El sistema muestra una interfaz con los profesores que no tienen proyecto asignado, dando la posibilidad de buscar profesores para asignarle el proyecto. El caso de uso finaliza cuando se emite el resultado de la opción solicitada.
Referencias	R87
Precondiciones	El Moderador se encuentra autenticado.
Poscondiciones	Es asignado el proyecto al profesor.

Tabla 6. Descripción del caso de uso Eliminar Profesor-Proyecto. (Descripción ampliada: Ver Anexo 2)

Caso de Uso	
CU-31	Eliminar Profesor-Proyecto.
Actores	Moderador (inicia).

Resumen	El caso de uso inicia cuando el Moderador escoge Eliminar Profesor-Proyecto. El sistema muestra un listado con los profesores y sus proyectos asignados respectivamente, dando la posibilidad de realizar búsqueda para suprimir alguna relación solicitada. El caso de uso finaliza cuando se emite el resultado de la opción solicitada.
Referencias	R88
Precondiciones	El Moderador se encuentra autenticado.
Poscondiciones	Es eliminada la relación del profesor con el proyecto.

Tabla 7. Descripción del caso de uso Asignar Estudiante-Proyecto. (Descripción ampliada: Ver Anexo 2)

Caso de Uso	
CU-34	Asignar Estudiante-Proyecto.
Actores	Moderador (inicia).
Resumen	El caso de uso inicia cuando el Moderador escoge Asignar Estudiante-Proyecto. El sistema muestra una interfaz con los estudiantes que no tienen proyecto asignado, dando la posibilidad de buscar estudiantes para asignarle el proyecto. El caso de uso finaliza cuando se emite el resultado de la opción solicitada.
Referencias	R91
Precondiciones	El Moderador se encuentra autenticado.
Poscondiciones	Es asignado el proyecto al estudiante.

Tabla 8. Descripción del caso de uso Eliminar Estudiante-Proyecto. (Descripción ampliada: Ver Anexo 2)

Caso de Uso	
CU-35	Eliminar Estudiante-Proyecto.
Actores	Moderador (inicia).
Resumen	El caso de uso inicia cuando el Moderador escoge Eliminar Estudiante-Proyecto. El sistema muestra la interfaz con los estudiantes y su proyecto correspondiente, dando la posibilidad de buscar para liberar a un estudiante de su proyecto correspondiente. El caso de uso finaliza cuando se emite el resultado de la opción solicitada.
Referencias	R92
Precondiciones	El Moderador se encuentra autenticado.
Poscondiciones	Es liberado un estudiante de su proyecto correspondiente.

2.9 Vista de Casos de Uso.

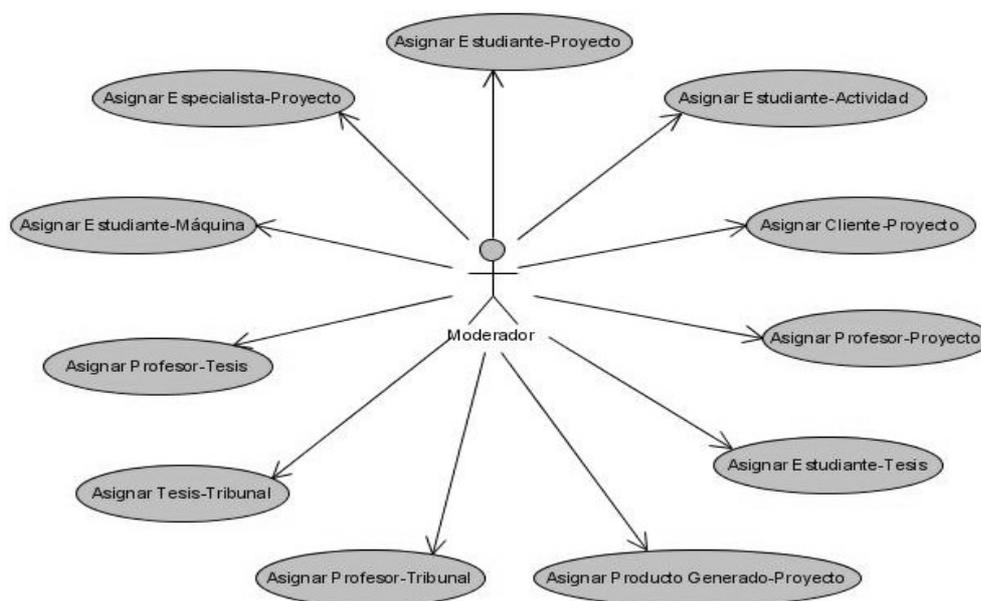


Figura 12. Casos de uso arquitectónicamente significativos.

2.10 Conclusiones.

En este capítulo se realizó la modelación del sistema a través del Modelo de Objetos del Dominio que permitió comprender el problema. Quedaron definidas 7 reglas del negocio, 123 requisitos funcionales y los requisitos no funcionales que debe tener la aplicación, 4 actores y 59 casos de uso que participan en el sistema, dichos casos de uso agrupados por paquetes y con sus descripciones textuales correspondientes.

CAPÍTULO 3

DESCRIPCIÓN DE LA SOLUCIÓN PROPUESTA

Introducción.

En este capítulo se describe como se desarrolla la solución del sistema con la utilización del CMS Drupal como mejor alternativa para realizar el portal de la facultad y sus facilidades para integrarle un módulo o componente que cumple con las funcionalidades requeridas en los paquetes de casos de uso: Producción, Tesis, Extensión Universitaria y Horario-Máquina de Proyectos y de los casos de uso: Visualizar Datos Generales, Adicionar Estudiante, Editar Estudiante, Adicionar Profesor y Editar Profesor. Se describe la arquitectura del sistema, se detalla el funcionamiento de dicho componente y se muestra el diagrama de despliegue, los prototipos de interfaz de usuario, el mapa de navegación y el modelo físico de la base de datos.

3.1 Descripción de la Solución Propuesta.

El presente trabajo está enmarcado en el proceso de gestión de información correspondiente a la facultad 6 y específicamente en el desarrollo de un portal Web en el que se publique toda la información necesaria para mantener informada a la misma.

Después de realizar una profunda investigación se decide usar el CMS Drupal como solución al problema, este CMS posee un conjunto de módulos que proveen una gran variedad de funcionalidades que cumplen con la mayoría de los requisitos propuestos por los clientes.

Este sistema de gestión de contenidos se emplea como una de las políticas propuestas por la dirección de informatización de la universidad, es ampliamente usado a nivel internacional por la integridad y seguridad de la información que presenta, así como otras características importantes como diseño, temas visuales y usabilidad.

Por las facilidades que brinda para los desarrolladores, ya que en su implementación se pueden incluir diferentes módulos realizados por una gran comunidad que le da soporte; fue posible integrarle un componente desarrollado a partir de varias herramientas, librerías y motores de plantillas, que complementa los servicios que no incluye el CMS para gestionar toda la información de la facultad referente a temas como: tesis, producción e investigación, extensión universitaria y laboratorios de proyectos.

El componente está desarrollado con la integración de PatTemplate, PatForms, PatErrors y otras herramientas y librerías como PEAR.

¿Qué necesita?

El componente necesita para su funcionamiento de varios archivos y variables de entrada:

- Un archivo con los parámetros necesarios para la conexión a la base de datos: usuario, contraseña, servidor, base de datos y puerto.
- Un archivo de configuración en el cual van a estar relacionados todos los parámetros referentes al elemento en cuestión en la base de datos. Por ejemplo: nombre de la tabla en la base de datos, título para mostrar al cliente, llave primaria en la base de datos, los campos de dicho elemento con algunos detalles como: tipo de dato, cantidad de caracteres, mínimo o máximo valor en caso de ser número, si debe ser mostrado, si es requerido, si puede ser editado, entre otros. Y al final contiene las relaciones entre los campos descritos en el archivo y sus correspondientes en la base de datos. Los campos también pueden ser el resultado de una consulta previa a la base de datos. Para visualizar y comprender mejor el contenido de este archivo ver: Figura 33, Capítulo 4.
- Una variable que se declara antes de hacer el llamado de ejecución del componente. La variable debe contener la dirección física del template a través del cual el componente va a imprimir los resultados de la solicitud del usuario.

¿Cómo funciona?

El componente comienza a ejecutarse cuando se llama a uno de sus archivos a través de la función `require_once()` en PHP. El archivo contiene las funciones necesarias para acceder a la base de datos y realizar las operaciones indicadas por el usuario y luego mostrar el resultado a través del "template" especificado previamente. El archivo requiere de librerías como PEAR, de clases de acceso a datos como DB; herramientas como: `patForms`, `patErrors`, `patErrorManager` y el motor de plantillas `PatTemplate`. Primeramente crea un formulario con los campos especificados en el archivo de configuración de entrada, luego los valida (en caso de haber algún error muestra un mensaje y no continúa con su ejecución), posteriormente carga el template que se especifica según la dirección que se declaró en la variable de entrada y después integra el formulario creado al template. Luego en dependencia de si es una inserción o una selección se conecta a la base de datos y realiza la operación requerida, si en la página debe mostrar datos, los imprime en el formulario creado y con las

características especificadas en el template, en caso contrario muestra un mensaje que refleja la acción realizada.

¿Cómo muestra la información?

La información se muestra según las especificidades del template que se llama según lo que se va a mostrar. El template tiene una configuración estándar, solo se le especifican en los campos que va a tener: los nombres y los valores que va a visualizar, en el caso de los valores se pone entre llaves el nombre del campo que contiene ese valor en la base de datos. Los valores que se van a mostrar pueden ser específicos de una tabla de la base de datos o el resultado de una consulta, en caso de que en la consulta se especifique algún alias, en el template se pondría el nombre del alias declarado. Los templates no admiten código en lenguaje PHP pero si en HTML por lo que pueden configurarse opciones visuales a comodidad del cliente.

Con la introducción de dicho componente se elimina la necesidad de implementar clases y se realiza el modelo físico de la base de datos a partir de los objetos contenedores de información del diagrama de clases del dominio y las reglas del negocio.

El CMS en conjunto con el componente antes descrito, cumple todas las necesidades solicitadas por los clientes de la facultad, brindando la información necesaria para mantener a los estudiantes y profesores actualizados sistemáticamente.

3.2 Arquitectura del Sistema.

Arquitectura del CMS Drupal.

Para el diseño del portal se utilizó el CMS Drupal, el cual emplea el patrón arquitectónico Modelo Vista Controlador (Model-View-Controller-MVC), utilizado para diseñar aplicaciones con sofisticadas interfaces donde la lógica de interfaz de usuario cambia con más frecuencia que los almacenes de datos y la lógica de negocio.

Este modelo de arquitectura presenta varias ventajas:

- ✓ Hay una clara separación entre los componentes de la aplicación, lo cual permite implementarlos por separado.
- ✓ Hay un API muy bien definido, cualquiera que lo use podrá reemplazar el Modelo, la Vista o el Controlador sin aparente dificultad.
- ✓ La conexión entre el Modelo y sus Vistas es dinámica, se produce en tiempo de ejecución.

Este estilo se basa en separar el diseño de las aplicaciones en 3 capas diferentes:

Modelo: Es la representación específica de la información con la cual el sistema opera. La lógica de datos asegura la integridad de estos y permite derivar nueva información.

Vista: Presenta el modelo en un formato adecuado para interactuar, usualmente las páginas HTML y el código que provee de datos dinámicos a las mismas.

Controlador: Responde a eventos, usualmente acciones del usuario e invoca cambios en el modelo y probablemente en la vista.

A continuación se muestra una representación básica de las relaciones entre las capas del patrón MVC:

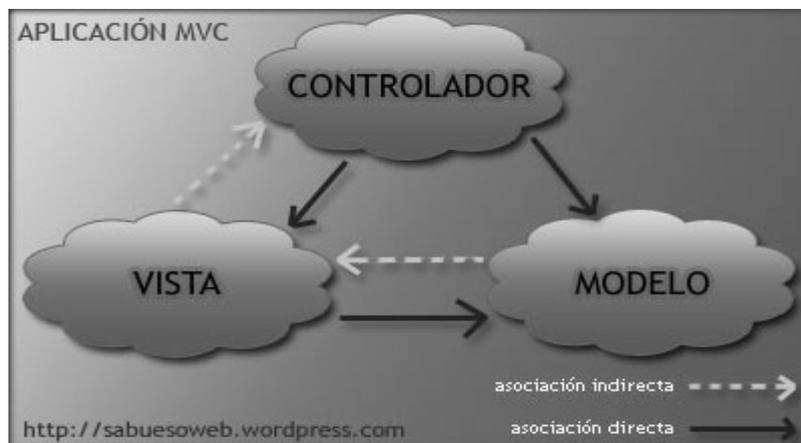


Figura 13. Representación del patrón MVC.

El objetivo de este patrón es realizar un diseño que desacople la vista del modelo, con la finalidad de mejorar la reusabilidad. De esta forma las modificaciones en las vistas impactan en menor medida en la lógica de negocio o de datos.

La siguiente figura la arquitectura del CMS dividida en las 3 capas correspondientes:

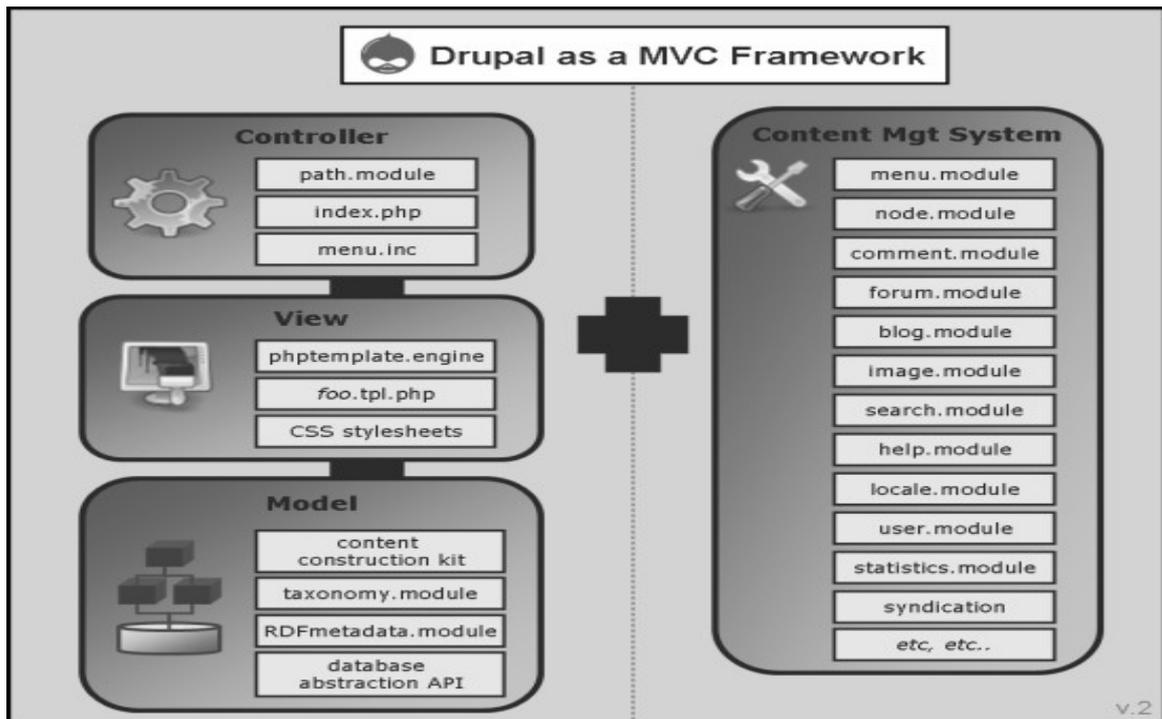


Figura 14. Representación del patrón MVC en Drupal.

Arquitectura del componente.

Para dar cumplimiento a un grupo de funcionalidades requeridas por los clientes no incluidas por el CMS Drupal, se le integra a la aplicación un componente implementado en lenguaje PHP. El componente utiliza como patrón arquitectónico 3 capas.

Capa de presentación o interfaz de usuario: Está formada por los elementos que intervienen en la presentación visual de información al usuario, es la capa que interactúa con el mismo.

Capa de negocio: Esta capa está formada por las entidades que representan objetos que van a ser manejados por toda la aplicación. En este caso, están representados por las clases, funciones, etc.

Capa de acceso a datos: Es donde residen los datos y es la capa encargada de acceder a los mismos. Está formada por uno o más gestores de base de datos que realizan todo el almacenamiento de datos y reciben solicitudes de almacenamiento o recuperación de información desde la capa de negocio.

La siguiente figura representa la estructura básica del patrón arquitectónico 3 capas.

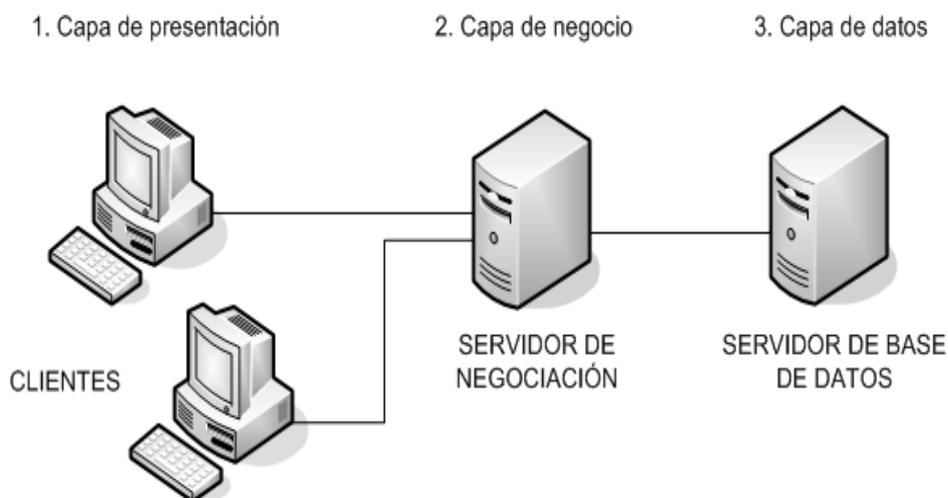


Figura 15. Representación del patrón 3 Capas.

¿Cómo se integran las arquitecturas MVC-3 Capas en el sistema?

La integración del componente al CMS se hace a través de los nodos (páginas) del CMS, que representan clases de la capa controlador, o sea, en las páginas (nodo X) se implementan instrucciones en lenguaje PHP que llaman a las clases (formulario.php, session.php, entre otras) pertenecientes a la capa de lógica del negocio en el componente y éstas son las encargadas de realizar determinadas operaciones, acceder a los datos (si es necesario) y finalmente mostrar la información a través de la capa (Vista) del CMS. En la figura que se muestra a continuación, se refleja gráficamente la integración de las dos arquitecturas.

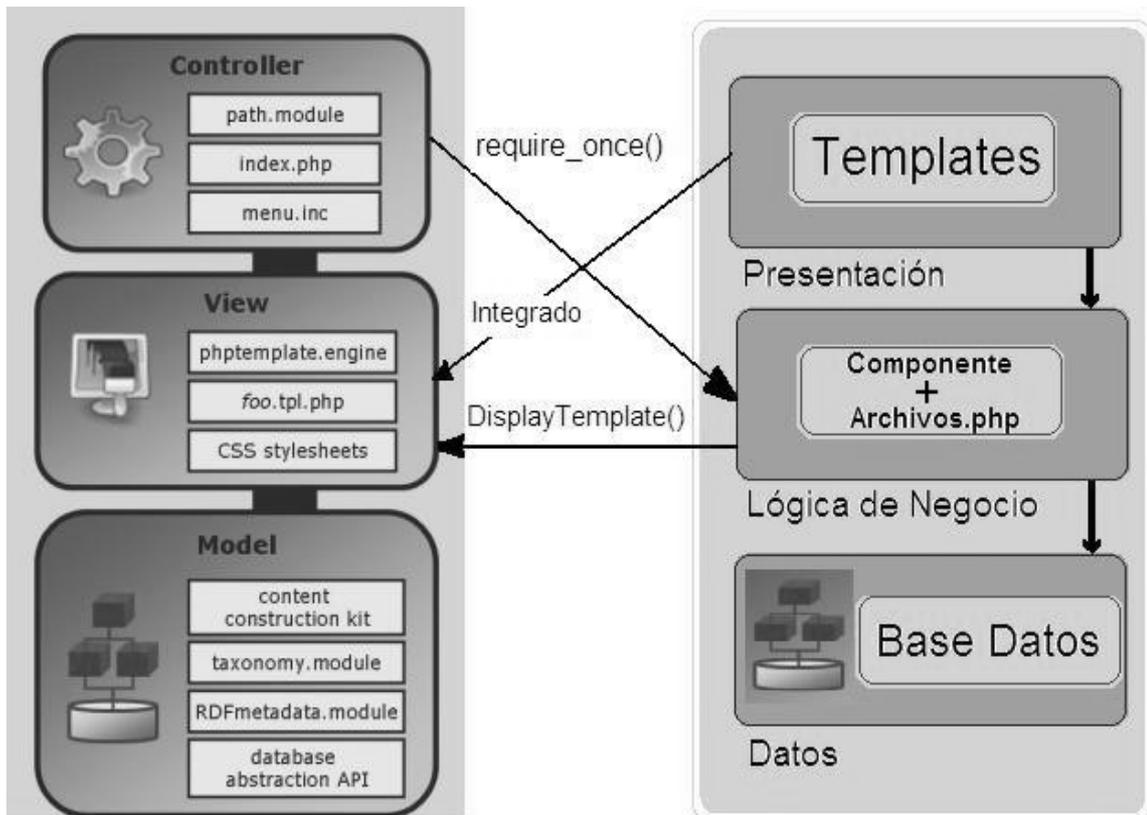


Figura 16. Representación de la integración de MVC y 3 Capas.

3.3 Vista de Despliegue.

El siguiente diagrama muestra las relaciones físicas entre los componentes hardware y software en el sistema final, es decir, la configuración de los elementos de procesamiento en tiempo de ejecución y los componentes software.

La aplicación necesita para su ejecución un servidor Web Apache, un servidor de base de datos MySQL, un servidor de dominio y las estaciones de trabajo, donde cada usuario tendrá acceso a la aplicación.

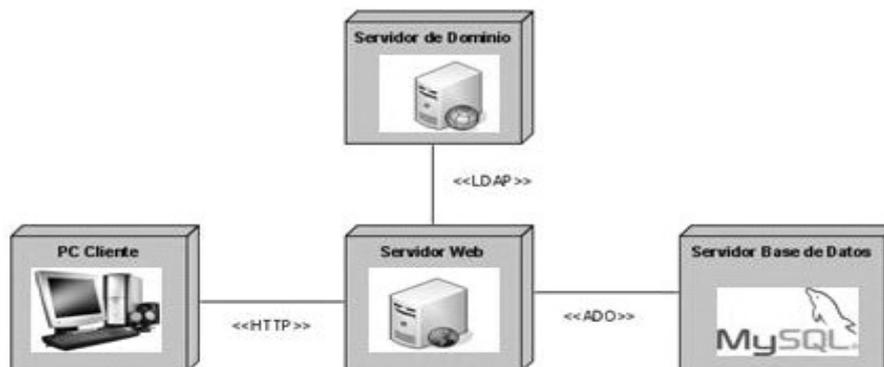


Figura 17. Diagrama de Despliegue.

3.4 Prototipos de Interfaz de Usuario.

Para la realización de los prototipos de interfaz de usuario el sistema integra un componente que utiliza el motor de plantillas PatTemplate y otras herramientas de PHP que a partir de plantillas predefinidas, construye las tablas y formularios correspondientes según las operaciones realizadas por el usuario.

A continuación se muestran imágenes de los prototipos no funcionales de algunos casos de uso significativos. Los prototipos restantes se encuentran en el expediente de proyecto.

Prototipos no funcionales correspondientes al caso de uso Visualizar Datos Generales.

Listado de Estudiantes.			
Filtrar:(Solapín/Nombre/Apellidos/Grupo/Apto)			
<input type="text"/>	<input type="button" value="Filtrar"/>		
Solapín	Nombre	Grupo	Apto
600000	Yanelis Romeu Ramos	6107	0
600001	Mercedes Batista Pérez	6201	
600002	Orlando Herrera González	6201	
600003	Yndira Inciarte Calderón	6201	
600004	Enrique Pérez Ramos	6201	
600005	Yasmany Santana Suárez	6201	

Figura 18. Prototipo no funcional CU- Visualizar Datos Generales: Escenario Visualizar Estudiantes.

Prototipos de interfaz de usuario de algunos casos de uso correspondientes al paquete Producción.

Adicionar/Modificar Proyecto.

Los campos con (*) son obligatorios.

Proyecto

Nombre* :

Fecha Inicio :

Fecha Fin :

Figura 19. Prototipo no funcional CU- Adicionar Proyecto.

Listado de Proyectos.

Filtrar:(Id_Proyecto/Nombre)

Id Proyecto	Nombre	Fecha Inicio	Fecha Fin	Opciones
2	Biosys	2008-03-25	2008-05-29	
5	MINPPAL	2008-01-01	2008-06-25	

Figura 20. Prototipo no funcional CU- Editar Proyecto.

Asignar Profesor-Proyecto

Los campos con (*) son obligatorios.

Asignar Proyecto a un Profesor

Identificador de Profesor* :

Identificador de Proyecto* :

Rol :

Figura 21. Prototipo no funcional CU- Asignar Profesor- Proyecto: Escenario Asignar Proyecto.

3.5 Mapa de Navegación.

El mapa de navegación refleja la estructura que tendrán las páginas web en la aplicación, representa la navegabilidad existente en el sistema. Existen muchas funcionalidades en la aplicación que se pueden agrupar siguiendo determinados criterios, por ejemplo: el contenido que gestionan, tipo de funcionalidad; por lo que se decide realizar varios mapas de navegación para facilitar la comprensión de la estructura del sistema. A continuación se representan los mapas de navegación correspondientes a las funcionalidades relacionadas con: Profesor-Estudiante, Extensión Universitaria y Horario-Máquina, los restantes mapas presentan un comportamiento similar y se encuentran en el expediente de proyecto.

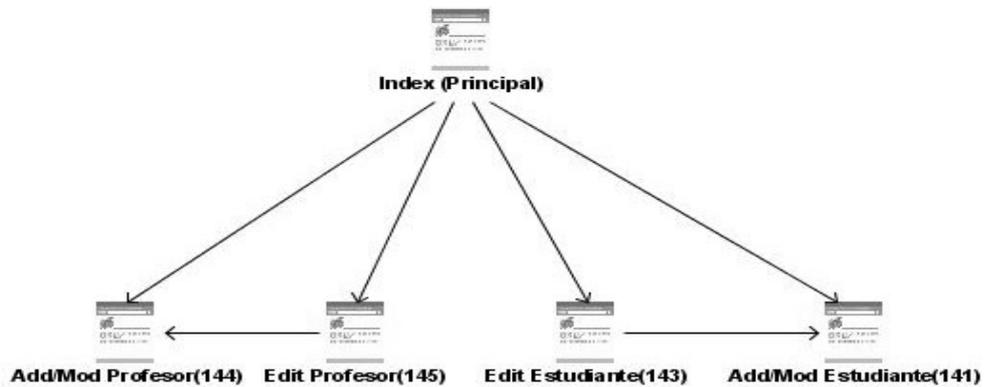


Figura 22. Mapa de Navegación correspondiente a Profesor-Estudiante.

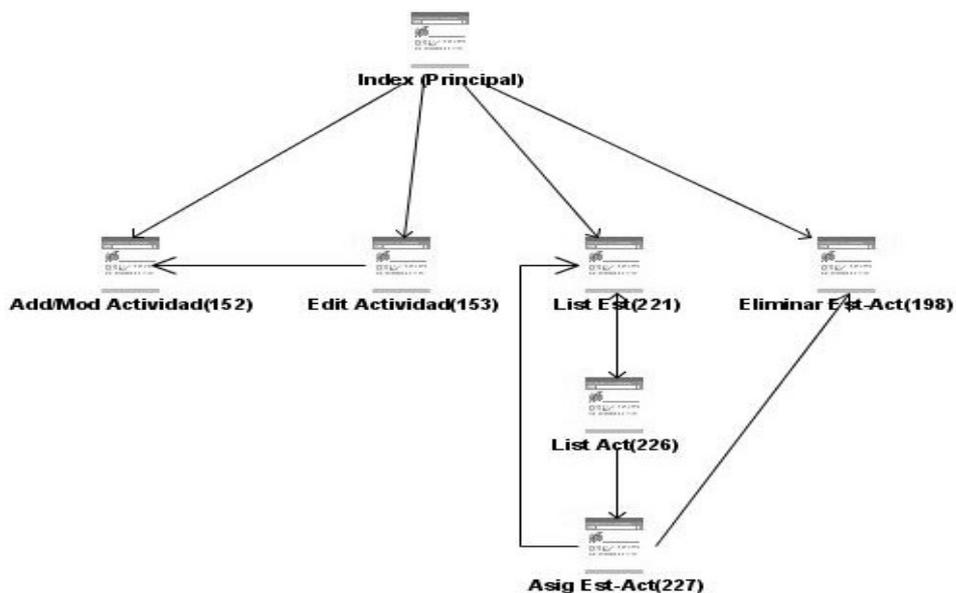


Figura 23. Mapa de Navegación correspondiente a Extensión Universitaria.

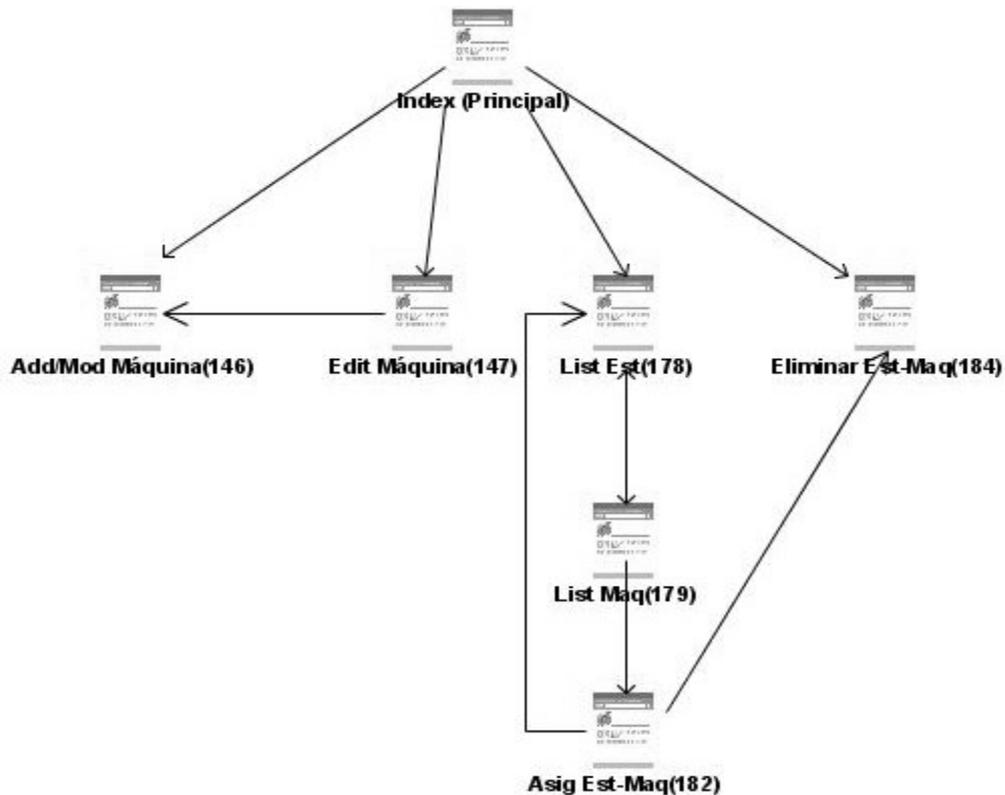


Figura 24. Mapa de Navegación correspondiente a Horario-Máquina.

3.6 Modelo Físico.

Partiendo de que los diagramas de clases del diseño y la realización de los casos de uso del diseño no se modelaron debido a la utilización de un componente para la implementación del sistema que implementa sus propias clases de acceso a datos y la generación de entidades en tiempo de ejecución; no fue necesario identificar las clases del diseño, razón por la cual no se obtuvo el diagrama de clases persistentes previo al modelado de la base de datos. Por tanto se realiza el modelo físico teniendo en cuenta como entradas: las clases contenedoras de información del diagrama de clases del modelo de dominio y las reglas del negocio. Las descripciones de las tablas de este modelo se encuentran en el expediente de proyecto. Dicho modelo se muestra a continuación:

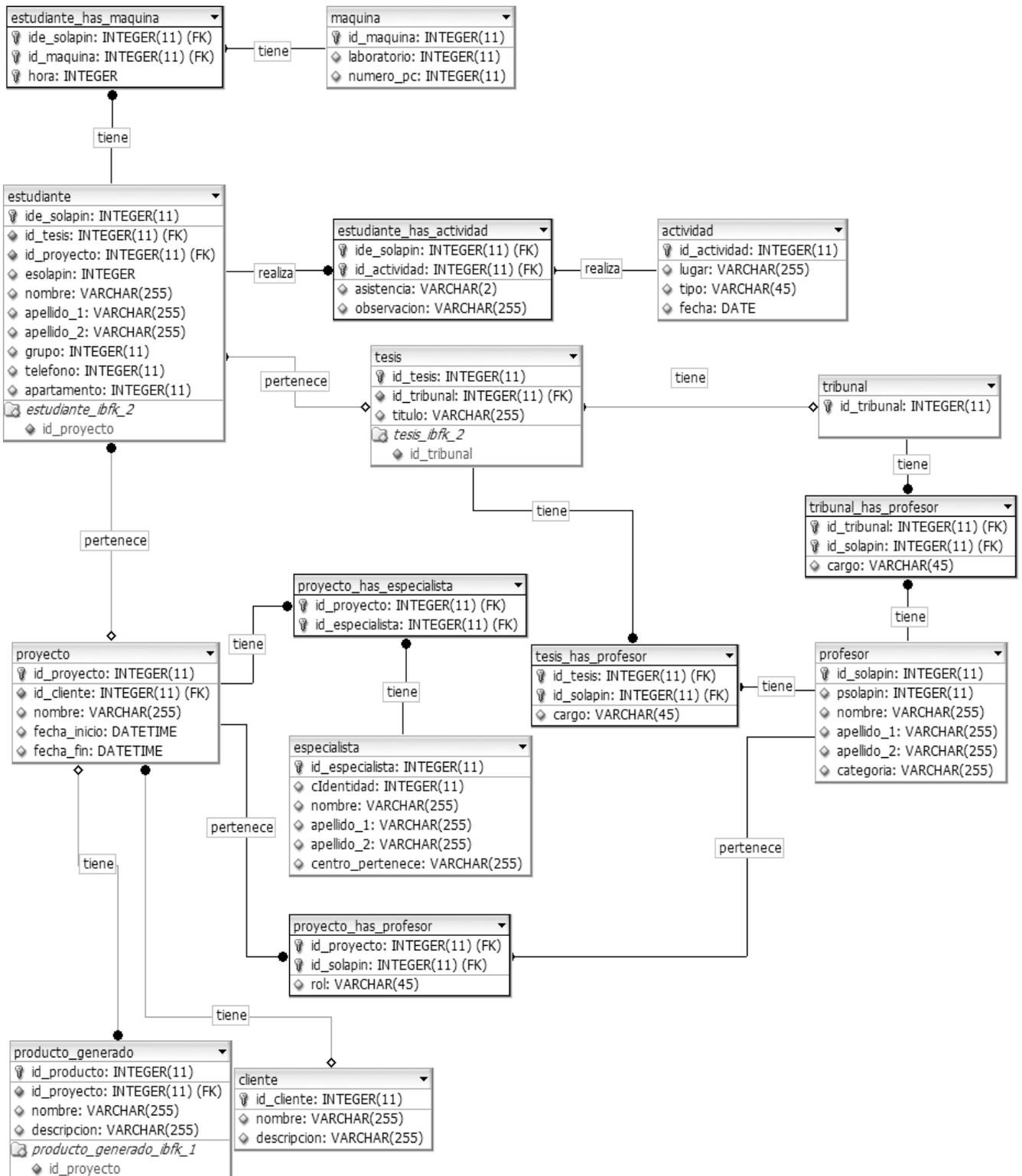


Figura 25. Modelo Físico.

3.7 Principios de Protección y Seguridad.

La seguridad en el sitio está implementada a través del servidor de base de datos MySQL y el uso de variables de sesión para restringir el acceso de los usuarios a determinadas páginas, existen varios niveles de seguridad, definidos como tipos de usuarios, teniendo en cuenta los roles que desempeñan los usuarios que interactúan con el sistema. El sistema posee un módulo de autenticación que verifica que el usuario está registrado y tiene acceso a la aplicación.

3.8 Conclusiones.

El uso del CMS Drupal para el diseño del portal ha facilitado considerablemente la integración de un componente al sistema que elimina la necesidad de una implementación significativa, lo que posibilita crear solamente pequeños archivos de configuración para su posterior funcionamiento. Además quedaron definidos el diagrama de despliegue, los prototipos no funcionales, el mapa de navegación y el modelo físico que brindan una mejor comprensión del funcionamiento de la aplicación.

CAPÍTULO IMPLEMENTACIÓN 4

Introducción.

En el presente capítulo se modelan los artefactos correspondientes al flujo de implementación. Se muestran fragmentos de código de los principales componentes y la descripción de su funcionamiento.

4.1 Diagrama de componentes.

“Los diagramas de componentes son usados para estructurar el modelo de implementación en términos de subsistemas de implementación y mostrar las relaciones entre los elementos de implementación.” [15]

A continuación se muestran los diagramas de componentes correspondientes a algunos de los casos de uso, los diagramas restantes expresan un comportamiento similar y se encuentran en el expediente de proyecto.

En este diagrama para cada visualización es un archivo "listar_lectura.php" con un código diferente, pero como tienen nombres iguales dejamos uno solamente en representación de los 10 que corresponden.

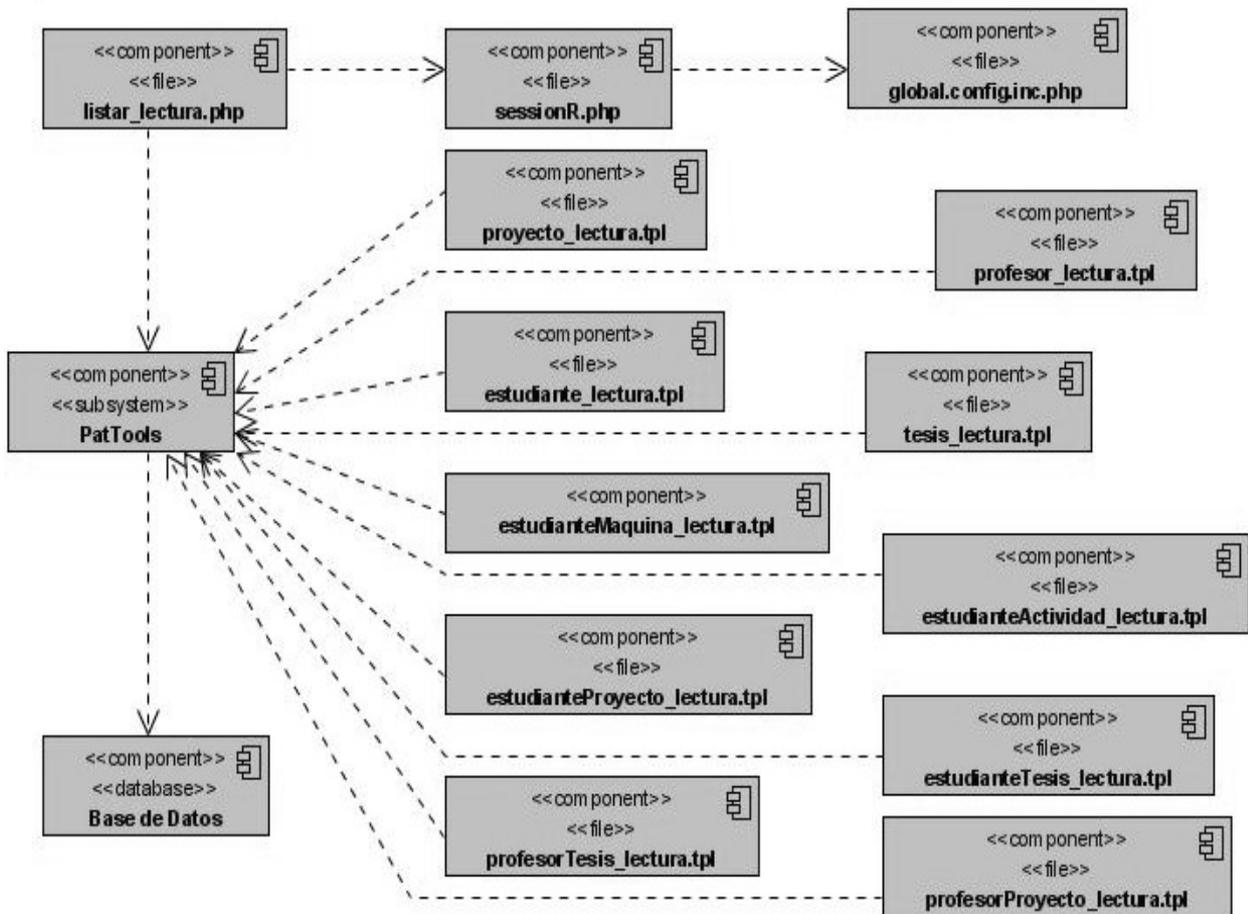


Figura 26. Diagrama de Componentes CU-Visualizar Datos Generales.

Diagramas de componentes correspondientes al paquete Producción.

Este diagrama es común para todos los casos de uso (Adicionar), solamente cambia el contenido del fichero config.php en el código por lo que no afecta dicho diagrama.

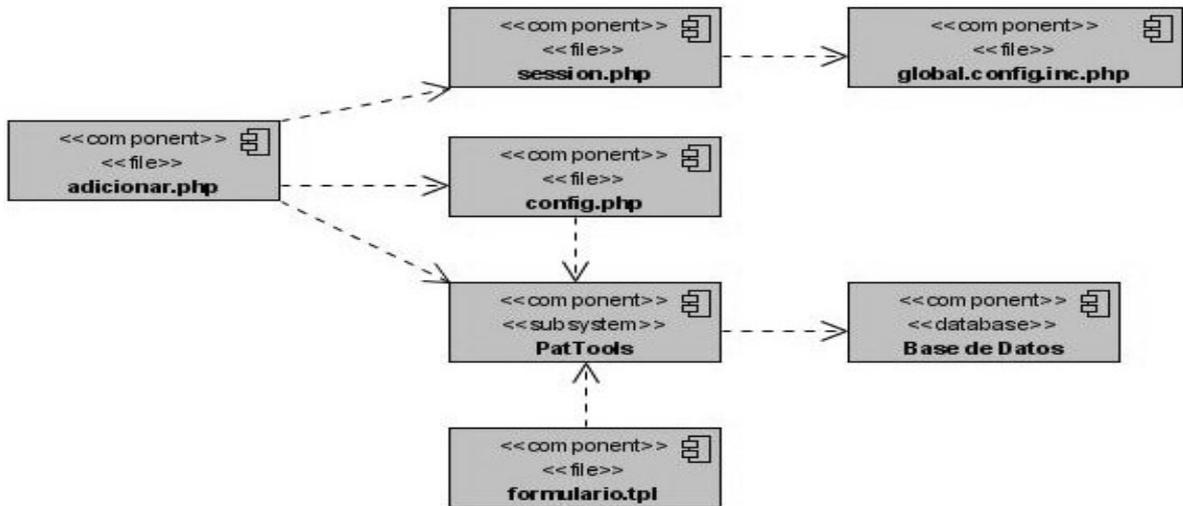


Figura 27. Diagrama de Componentes CU-Adicionar Proyecto.

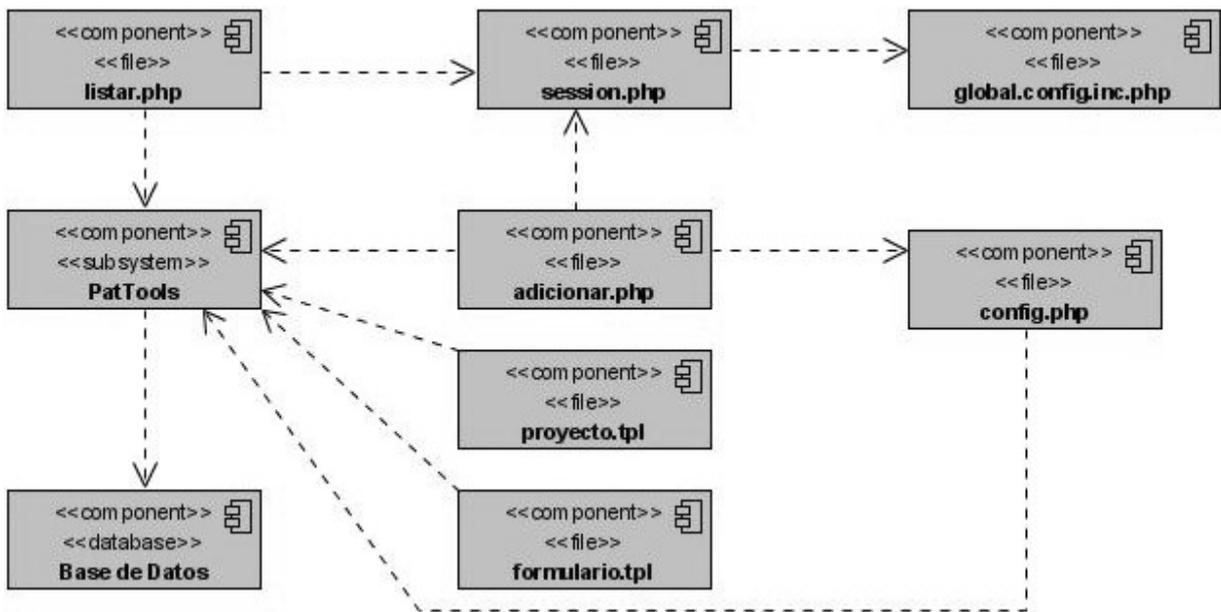


Figura 28. Diagrama de Componentes CU-Editar Proyecto

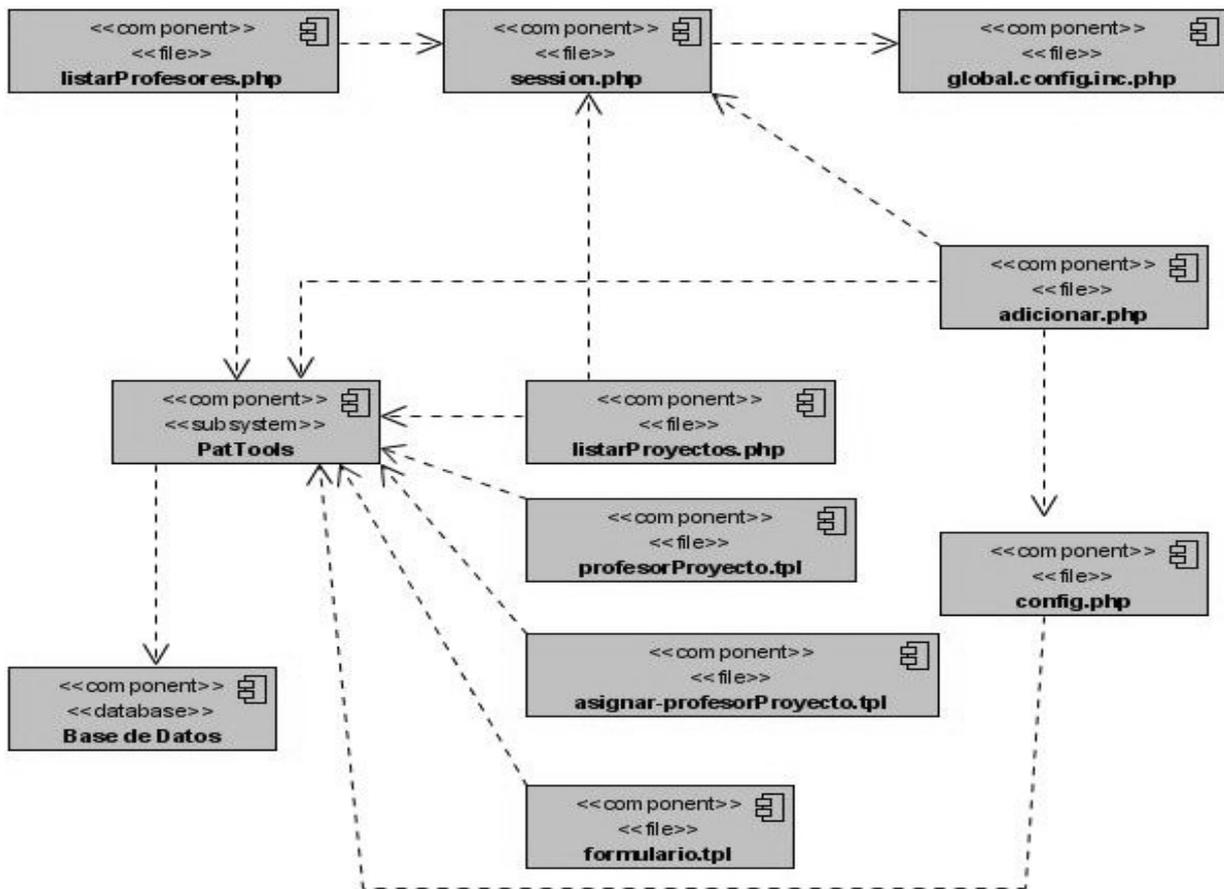


Figura 29. Diagrama de Componentes CU-Asignar Profesor- Proyecto.

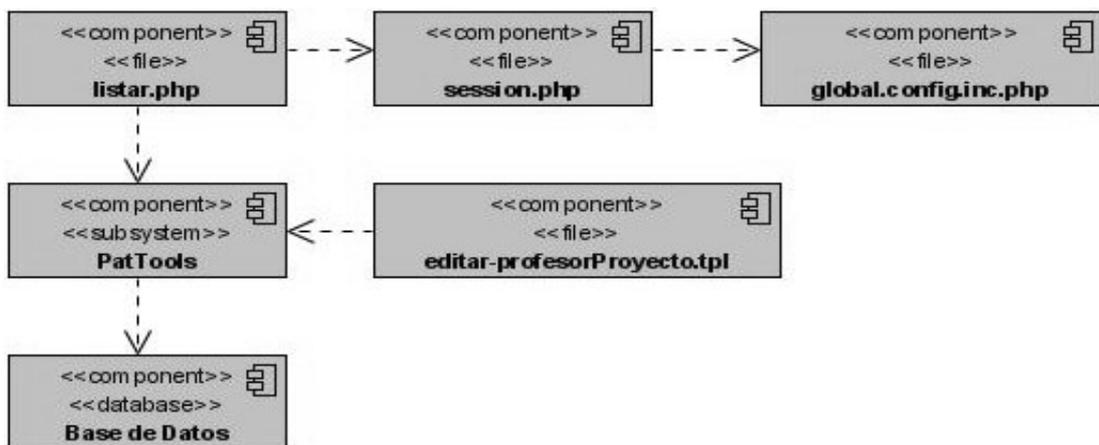


Figura 30. Diagrama de Componentes CU-Eliminar Profesor- Proyecto.

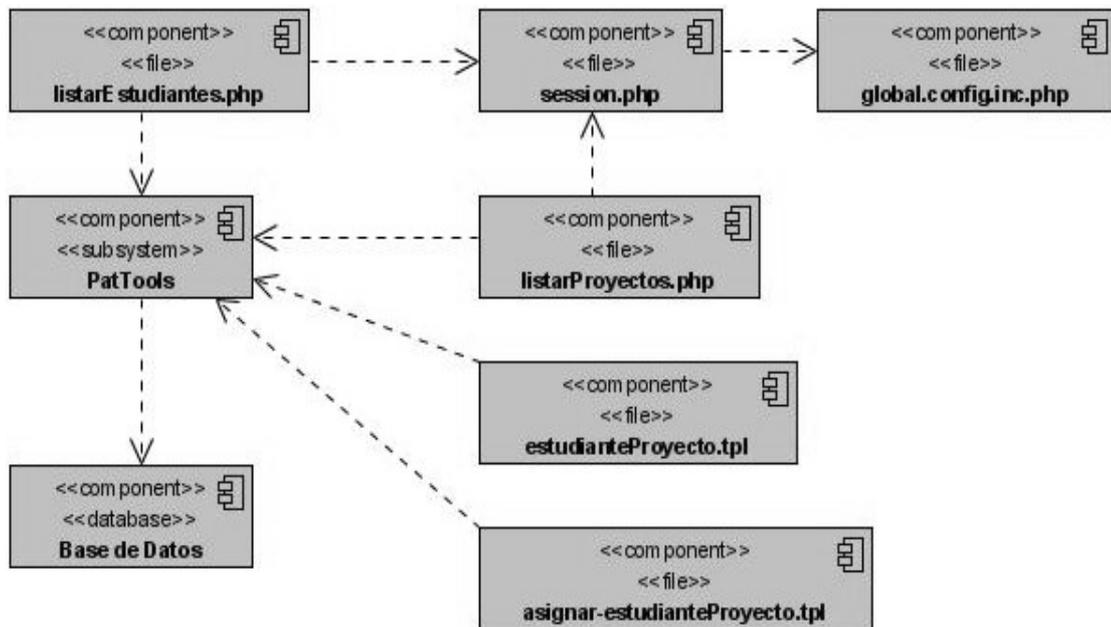


Figura 31. Diagrama de Componentes CU-Asignar Estudiante- Proyecto.

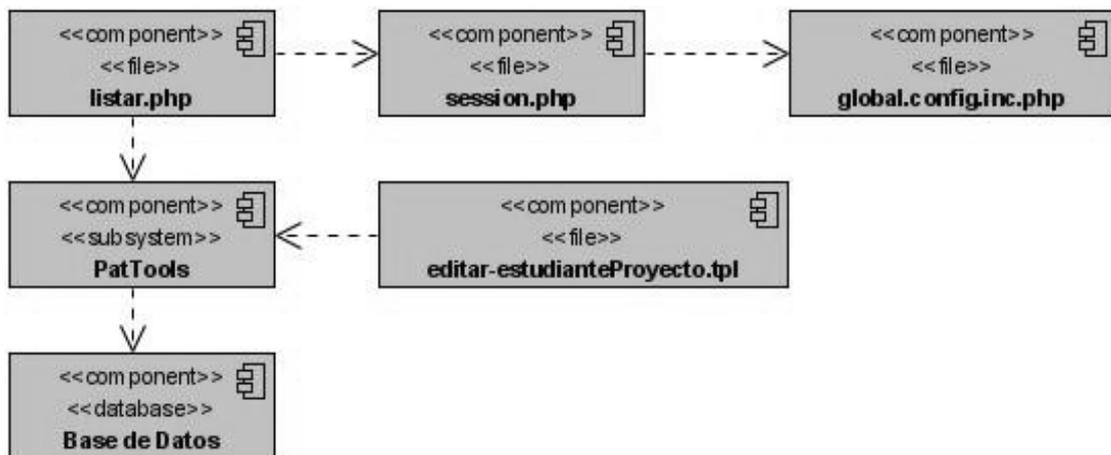


Figura 32. Diagrama de Componentes CU-Eliminar Estudiante- Proyecto.

4.2 Código fuente de los principales componentes y descripciones de los mismos.

El código que se muestra y describe a continuación se corresponde con los componentes implementados de mayor importancia, los demás no se detallan por no tener un gran significado en el código o por ser muy sencillos a la hora de implementarlos.

```

1  <?php
2  require_once("conexion.php");
3  $table      = 'tesis';
4  $titulo     = 'Tesis';
5  $pkey       = 'id_tesis';
6
7  $elements = array(
8
9      'id_tesis' => array(
10         'type'      => 'Number',
11         'attributes' => array(
12             'required' => 'no',
13             'display'  => 'no',
14             'edit'     => 'no',
15             'id'       => 'id_tesis',
16             'default'  => $_REQUEST['id_tesis'],
17         ),
18     ),
19
20     'titulo' => array(
21         'type'      => 'String',
22         'attributes' => array(
23             'required' => 'yes',
24             'display'  => 'yes',
25             'edit'     => 'yes',
26             'id'       => 'titulo',
27             'label'    => 'Título*',
28             'title'    => 'Título',
29         ),
30     ),
31
32 );
33
34
35 $FieldMap = array (
36     'id_tesis' => 'id_tesis',
37     'titulo'   => 'titulo',
38 );
39 >>

```

Figura 33. Código correspondiente al fichero “config.php” del CU Adicionar Tesis.

La figura 33 muestra el código correspondiente a uno de los ficheros implementados, en este caso “config.php” del CU Adicionar Tesis. El código está escrito en lenguaje PHP y representa una entrada en forma de ORM (**Object-Relational Mapping**) bastante simple para un componente funcional del sistema. El código consiste en declarar un conjunto de parámetros y sus campos homólogos correspondientes en la base de datos, así como diferentes características que deben poseer estos campos para su posterior manejo en tiempo de ejecución (validaciones).

Para detallar este fragmento de código podemos apreciar que en las primeras líneas se requiere del archivo conexion.php que no es más que un archivo de conexión a la base de datos propio del componente. Posteriormente se declaran varios campos como son: nombre de la tabla en la base de datos, título con el que se mostraría esa entidad en el cliente y la llave primaria de esa tabla en la base de datos.

Luego se declara un arreglo asociativo con todos los campos pertenecientes a esa entidad y su correspondencia en la base de datos.

Para cada campo se describe el tipo de dato, si es requerido/mostrado o no, si puede ser editable, el nombre con que se muestra al cliente, su valor por defecto (opcional) y se declaran también otras opciones como valores mínimos/máximos, cantidad de caracteres, entre otras.

Por último se declara otro arreglo asociativo donde se define cada campo de los declarados anteriormente a que campo corresponde en la base de datos real.

```

1  <?php
2      require_once('modulos/sessionR.php');
3      require_once("conexion.php");
4
5      if ( $_REQUEST['filtro'] != null ) {
6
7          $filtro=$_REQUEST['filtro'];
8          $query = "select `estudiante`.`ide_solapin`,`nombre`,`apellido_1`,`apellido_2`,
9                  `maquina`.`id_maquina`,`laboratorio`,`numero_pc`,`hora`
10         from `estudiante` join `estudiante_has_maquina` join `maquina`
11         on `estudiante`.`ide_solapin`=`estudiante_has_maquina`.`ide_solapin`
12         and `maquina`.`id_maquina`=`estudiante_has_maquina`.`id_maquina`
13         and (`nombre` like '%%$filtro%' or `apellido_1` like '%%$filtro%' or `apellido_2` like '%%$filtro%'
14         or `laboratorio` like '%%$filtro%' or `numero_pc` like '%%$filtro%' or `hora` like '%%$filtro%')";
15
16     }else{
17         $query = "select `estudiante`.`ide_solapin`,`nombre`,`apellido_1`,`apellido_2`,
18                 `maquina`.`id_maquina`,`laboratorio`,`numero_pc`,`hora`
19         from `estudiante` join `estudiante_has_maquina` join `maquina`
20         on `estudiante`.`ide_solapin`=`estudiante_has_maquina`.`ide_solapin`
21         and `maquina`.`id_maquina`=`estudiante_has_maquina`.`id_maquina`";
22     }
23
24     $template = 'estudianteMaquina_lectura.tpl';
25
26     require_once('listar_lib.php');
27  ?>

```

Figura 34. Código correspondiente al fichero “listar_lectura.php” del CU Visualizar Datos Generales: Escenario Visualizar Estudiantes.

La figura 34 muestra el código correspondiente a uno de los ficheros implementados, en este caso “listar_lectura.php” del CU Visualizar Datos Generales: Escenario Visualizar Estudiantes. El código está escrito en lenguaje PHP y su objetivo principal es declarar dos parámetros necesarios para un componente funcional del sistema que los utiliza para su posterior funcionamiento. El código consiste en asignar a una variable llamada \$query las diferentes consultas posibles a la base de datos cuyos resultados luego serían mostrados al cliente. Además se define una variable \$template con un valor que el componente mencionado interpretaría como dirección a un fichero.

Para detallar este fragmento de código podemos apreciar que en las primeras líneas se requiere de dos ficheros, uno es el que trabaja con las sesiones y su nombre lo representa (sessionR.php) y el otro (conexion.php) es un fichero del componente que se encarga de la conexión con la base de datos.

Luego se muestra la declaración de la variable referente a la consulta deseada según las acciones del cliente, en el caso de que el cliente entre un criterio para filtrar los datos se asignará a la variable \$query la primera variante y en caso contrario se mostrará directamente el resultado de la segunda consulta predefinida.

También se define una variable \$template con un valor que será interpretado posteriormente como una dirección de un fichero de entrada para el componente funcional.

4.3 Conclusiones

En este capítulo se realizaron 34 diagramas de componentes que mostraron por su baja complejidad una fácil integración del sistema, también se describieron algunos fragmentos de códigos de los principales componentes implementados que demuestran fácilmente la adaptabilidad e independencia del código con el sistema, lo que permite reutilizar dicho código en otras aplicaciones con características similares.

CONCLUSIONES

Como resultado del desarrollo del Portal Web de la Facultad 6 se puede concluir que:

- Se estudiaron los procesos de negocio en cada área de la facultad, lo que permitió identificar las principales necesidades a resolver con el desarrollo de esta aplicación.
- La Base Datos implementada permitió registrar toda la información requerida por las diferentes áreas de la facultad.
- Los datos manejados por la facultad son mostrados a través de diferentes servicios, que facilitan las vías de acceso a estos de forma rápida.
- Con la incorporación de un nuevo componente se logró abarcar muchas de las necesidades requeridas por los clientes.

RECOMENDACIONES

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

- La realización de las pruebas a la aplicación.
- La implementación del horario docente de forma dinámica para comodidad de los estudiantes y profesores.
- Se recomienda seguir el estudio de las potencialidades que brinda el CMS Drupal para implementar nuevas funcionalidades en dependencia de las necesidades de la facultad.
- Migrar las bases de datos usadas por la aplicación a PostgreSQL, por las limitaciones que para Cuba trajo que MySQL fuera adquirido por la Compañía Sun Microsystems en enero del 2008.
- La utilización del presente trabajo como bibliografía para posibles investigaciones referentes al tema desarrollado en el mismo.

REFERENCIAS BIBLIOGRÁFICAS

1. **Arceo, Claro, Alfonso y Sánchez, Pantoja, Maidolys.** *“Portal Digital de la Facultad Regional de la UCI en Granma”*. La Habana : s.n., 2007. Tesis de diploma.
2. **Informática Milenium.** [En línea] 1998_2006. [Citado el: 2 de noviembre de 2007.] <http://www.informaticamilenium.com.mx/paginas/espanol/sitioweb.htm>.
3. **Lago, Díaz, César. Webmaster.** *Administrador de Sistemas Centro de Innovación de la Universidad de Oviedo Campus de Mieres*. s.l. : 6 de noviembre de 2007.
<http://www.uniovi.es>
4. **Durón, Aguirre, Francisco Javier.** Ingeniero de la Universidad Regiomontana. s.l. : 6 de noviembre de 2007.
<http://www.ur.mx>
5. **Izquierdo, Luis. Webmaster.** Facultad de Matemática y Computación. s.l. : 8 de noviembre de 2007.
<http://www.matcom.uh.cu>
6. **Cuerda, García, Xavier y Alfonso, Minguillón, Juliá.** *“Introducción a los Sistemas de Gestión de Contenidos (CMS) de código abierto.”* 2004. [En línea] [Citado el: 8 de noviembre de 2007.]
<http://mosaic.uoc.edu/articulos/cms1204.html>
7. **Álvarez, Miguel Ángel.** DesarrolloWeb. *“Introducción a Java Script.”* [En línea] [Citado el: 10 de noviembre de 2007.]
<http://www.desarrolloweb.com/articulos/490.php>.
8. **PHP.** [En línea] 2007. [Citado el: 10 de noviembre de 2007.]
<http://www.php.net/manual/es/introduction.php>.
9. **Álvarez, Miguel Ángel.** DesarrolloWeb. *“Qué es PHP”*. [En línea] [Citado el: 10 de noviembre de 2007.] <http://www.desarrolloweb.com/articulos/392.php>.
10. **Sánchez, Mendoza, María A.** *“Metodologías De Desarrollo De Software”*. [En línea] 7 de junio de 2004. [Citado el: 15 de noviembre de 2007.]
http://www.informatizate.net/articulos/pdfs/metodologias_de_desarrollo_de_software_07062004.pdf.

11. **Jacobson, I, Booch, G y Rumbaugh, J.** *“El Proceso Unificado de Desarrollo de Software”*. La Habana : Félix Varela, 2004. Vol. 1.
12. Ingeniería de Software 1. *“Introducción a la Ingeniería de Software.”* UCI. curso 2007_2008. Conferencia #1.
13. **Jacobson, I, Booch, G y Rumbaugh, J.** *“El Proceso Unificado de Desarrollo de Software.”* La Habana : Félix Varela, 2004. Vol. 1.
14. Ingeniería de Software 1. *“Flujo de Trabajo de Requerimientos.”* UCI. curso 2007_2008. Conferencia #4.
15. Ingeniería de Software 2. *“Flujo de Implementación .”* UCI. curso 2006_2007. Conferencia.

BIBLIOGRAFÍA

1. **Álvarez, Miguel Ángel.** Desarrollo Web. “*Qué es HTML.*” [En línea][Consultado el: 19 noviembre, 2007].
<http://www.desarrolloweb.com/articulos/534.php>
2. **Álvarez, Miguel Ángel.** Desarrollo Web. “*MySQL Query Browser.*” [En línea][Consultado el: 22 mayo, 2008].
<http://www.desarrolloweb.com/articulos/1832.php>
3. **Arceo, Claro, Alfonso y Sánchez, Pantoja, Maidolys.** “*Portal Digital de la Facultad Regional de la UCI en Granma.*” La Habana : s.n., 2007. Tesis de diploma.
4. **Arquitectura Modelo/Vista/Controlador** [En línea][Consultado el: 10 junio, 2008].
http://www.cica.es/formacion/JavaTut/Apendice/arq_mvc.html
5. **Batista, Torres, Nelliud D.** “*El uso de Portales Corporativos en la Gerencia del Conocimiento para las Empresas.*” Gurabo, Puerto Rico : s.n., 2006. Programa Doctoral en Administración de Empresas.
6. **Delgado, Alonso, Yanet y Mulet, González, Yunet.** “*Software para la Simulación de Sistemas Biológicos: Módulo de Simulación y Análisis*” La Habana : s.n., 2007. Tesis de diploma.
7. **Gracia, Joaquín.** “*Introducción a MySQL.*” Última actualización: 9 de junio del 2005. [Consultado el: 19 de noviembre, 2007]. Disponible en:
<http://www.webestilo.com/mysql/>
8. **Hernández, Rodríguez, Yunior y Moret, Vargas, Adalberto Gendry.** “*Diseño del Portal Web Bohemia.*” La Habana : s.n., 2007. Tesis de diploma.
9. **Herramientas Web para la enseñanza de protocolos de comunicación.** “*El modelo cliente - servidor.*” [Consultado el: 14 noviembre, 2007]. Disponible en:
<http://neo.lcc.uma.es/evirtual/cdd/tutorial/aplicacion/cliente-servidor.html>
10. **Jara, Sánchez, Jesús y Paterson, Douglas.** “*Construyendo sitios Web con PHP-Nuke.*” [Español] ,2007. [Consultado: 15 de noviembre, 2007]. Disponible en:
<http://www.packtpub.com/php-nuke-es/book>

11. **Joomla.** *¿Qué es Joomla?* [Consultado: 16 de noviembre, 2007]. Disponible en:
<http://www.joomlaos.net/>
12. **Kate.** [Consultado: 17 de mayo, 2008]. Disponible en:
<http://www.guia-ubuntu.org/index.php?title=Kate>
13. **MSF.** “*Introducción a la Ingeniería de Software.*” [En línea] Mayo de 2007. [Consultado el: 7noviembre, 2007]. <http://rrivera334.blogspot.es/1179459060/>
14. **Orallo, Hernández, Enrique.** “*El Lenguaje Unificado de Modelado (UML).*” [En línea] [Consultado el: 6 noviembre, 2007].
<http://www.disca.upv.es/enheror/pdf/ActaUML.PDF>.
15. **Parada, Plá, Indira.** “*Portal digital para el Centro de Control Estatal de Equipos Médicos.*” La Habana : s.n., 2007. Tesis de diploma.
16. **Perodín, Sánchez, Yusdenis y Moreno, Armentero, Yuniesky.** “*Software para la Simulación de Sistemas Biológicos: Módulo de modelación gráfica de Sistemas Biológicos*” La Habana : s.n., 2007. Tesis de diploma.
17. **Post Nuke.** *¿Qué es PostNuke?* [Consultado: 15 de noviembre, 2007]. Disponible en:
<http://www.cmsenespanol.com/modules/smartsection/item.php?itemid=45>
18. **Programación en castellano.** “*Visual Paradigm for UML.*” Última actualización: 5 de julio de 2005. [Consultado el: 8 de noviembre, 2007]. Disponible en:
<http://www.programacion.com/noticia/1363/>
19. **Rational.** *Rational Unified Process.* 2003.
20. **Reyero, José A.** *Comunidad de usuarios de Drupal.* En línea desde el 11 abril 2005. [Consultado: 18 de noviembre, 2007].
<http://www.drupal.org.es/caracteristicas>
21. **Robertson, James.** “*So, what is a content management system?*” 2005. [Consultado: 5 de noviembre, 2007]. Disponible en:
http://www.steptwo.com.au/papers/kmc_what/index.html

22. **Rodríguez, González, Grissel Belkis y Casals, Muñoz, Velmour.** “*Desarrollo del Portal Digital de la Facultad Regional Mártires de Artemisa.*” La Habana : s.n., 2007. Tesis de diploma.
23. **Sarduy, Domínguez, Y y Urra, González, P.** “*Sistemas de gestión de contenidos: en busca de una plataforma ideal.*” 2006. [Consultado: 5 de noviembre, 2007]. Disponible en:
http://bvs.sld.cu/revistas/aci/vol14_4_06/aci11406.htm
24. **Una Introducción a Apache.** 2006. [Consultado el: 19 de noviembre, 2007]. Disponible en:
http://linux.ciberaula.com/articulo/linux_apache_intro/
25. **Valdés, Pérez, Damián.** “*Editores Web que facilitan tu trabajo.*” 2007. Disponible en:
<http://www.maestrosdelweb.com/editorial/editores-web-que-facilitan-tu-trabajo/>

ANEXOS

Anexo1: Arquitectura Cliente-Servidor.

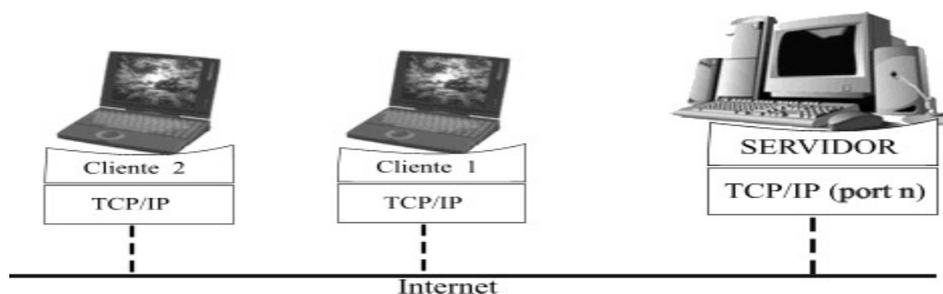


Figura 35. Representación de la arquitectura Cliente-Servidor.

Anexo 2: Descripciones ampliadas de los casos de uso (Asignar Profesor-Proyecto, Eliminar Profesor-Proyecto, Asignar Estudiante-Proyecto, Eliminar Estudiante-Proyecto).

Tabla 9. Descripción ampliada del caso de uso Asignar Profesor-Proyecto.

Caso de Uso	
CU-30	Asignar Profesor-Proyecto
Actores	Moderador (inicia).
Resumen	El caso de uso inicia cuando el Moderador escoge Asignar Profesor-Proyecto. El sistema muestra una interfaz con los profesores que no tienen proyecto asignado, dando la posibilidad de buscar profesores para asignarle el proyecto. El caso de uso finaliza cuando se emite el resultado de la opción solicitada.
Referencias	R87
Precondiciones	El Moderador se encuentra autenticado.
Poscondiciones	Es asignado el proyecto al profesor.
Flujo Normal de Eventos	

Acción del Actor	Respuesta del Sistema
<p>1. El Moderador desea asignar proyecto a profesor.</p> <p>3. El Moderador desea realizar una de las siguientes opciones:</p> <ul style="list-style-type: none"> • Buscar profesores definiendo criterio. • Seleccionar proyecto. 	<p>2. El sistema muestra una interfaz con todos los profesores que no tienen proyecto asignado y opciones para filtrar y/o seleccionar el proyecto.</p> <p>4. El sistema en dependencia de la opción que solicita realizar el Moderador, ejecuta las siguientes acciones:</p> <p>a) Si decide buscar profesores. Ver sección: “Buscar Profesores”.</p> <p>b) Si decide seleccionar proyecto. Ver sección: “Seleccionar Proyecto”.</p>
Sección : “Buscar Profesores”	
Acción del Actor	Respuesta del Sistema
<p>1. Introduce uno de los parámetros por el cual desea realizar la búsqueda:</p> <ul style="list-style-type: none"> • Solapín • Nombre • Apellidos <p>2. El Moderador presiona el botón “Filtrar”.</p>	<p>3. El sistema muestra un listado según el parámetro seleccionado por el Moderador.</p> <p>4. Finaliza el CU.</p>
Flujo Alternativo de la Sección : “Buscar Profesores”	
Acción del Actor	Respuesta del Sistema
	<p>3.1 Si el sistema no encuentra profesores con esas características muestra un listado vacío. Ir a la</p>

	acción1.
Sección : “Seleccionar Proyecto”	
Acción del Actor	Respuesta del Sistema
<p>1. El Moderador escoge el profesor que desea asignarle el proyecto y presiona la opción Selecc.Proyecto.</p> <p>3. El Moderador desea realizar una de las siguientes opciones:</p> <ul style="list-style-type: none"> • Buscar proyectos definiendo criterio. • Asignar Proyecto. 	<p>2. El sistema muestra una interfaz con todos los proyectos que no tienen profesores asignados y opciones para filtrar y/o asignar el proyecto.</p> <p>4. El sistema, en dependencia de la opción que solicita realizar el Moderador, ejecuta las siguientes acciones:</p> <p>a) Si decide buscar proyectos. Ver sección: “Buscar Proyectos”.</p> <p>b) Si decide asignar proyecto. Ver sección: “Asignar Proyecto”.</p>
Sección : “Buscar Proyectos”	
Acción del Actor	Respuesta del Sistema
<p>1. Introduce uno de los parámetros por el cual desea realizar la búsqueda:</p> <ul style="list-style-type: none"> • Nombre • Fechas <p>2. El Moderador presiona el botón “Filtrar”.</p>	<p>3. El sistema muestra un listado según el parámetro seleccionado por el Moderador.</p> <p>4. Finaliza el CU.</p>
Flujo Alternativo de la Sección : “Buscar Proyectos”	
Acción del Actor	Respuesta del Sistema

	3.1 Si el sistema no encuentra proyectos con esas características muestra un listado vacío. Ir a la acción1.
Sección : “Asignar Proyecto”	
Acción del Actor	Respuesta del Sistema
<p>1. El Moderador escoge el proyecto que desea asignarle al profesor y presiona la opción Asignar Proyecto.</p> <p>3. El Moderador presiona el botón “Enviar.”</p>	<p>2. El sistema muestra una interfaz con los datos previamente seleccionados.</p> <p>4. El sistema emite un mensaje y muestra los datos actualizados.</p> <p>5. Finaliza el CU.</p>
Prioridad	Crítico.

Tabla 10. Descripción ampliada del caso de uso Eliminar Profesor-Proyecto.

Caso de Uso	
CU-31	Eliminar Profesor-Proyecto.
Actores	Moderador (inicia).
Resumen	El caso de uso inicia cuando el Moderador escoge Eliminar Profesor-Proyecto. El sistema muestra un listado con los profesores y sus proyectos asignados respectivamente, dando la posibilidad de realizar búsqueda para suprimir alguna relación solicitada. El caso de uso finaliza cuando se emite el resultado de la opción solicitada.
Referencias	R88
Precondiciones	El Moderador se encuentra autenticado.

Poscondiciones	Es eliminada la relación del profesor con el proyecto.
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
<p>1. El Moderador desea eliminar la relación del profesor con el proyecto que tiene asignado.</p> <p>3. El Moderador desea realizar una de las siguientes opciones:</p> <ul style="list-style-type: none"> • Buscar profesor-proyecto definiendo criterio. • Eliminar profesor-proyecto específico. 	<p>2. El sistema muestra una interfaz con todos los profesores y proyectos existentes y opciones para filtrar y/o eliminar.</p> <p>4. El sistema en dependencia de la opción que solicita realizar el Moderador, ejecuta las siguientes acciones:</p> <p>a) Si decide buscar profesor-proyecto. Ver sección: "Buscar Profesor-Proyecto".</p> <p>b) Si decide eliminar profesor-proyecto. Ver sección: "Eliminar Profesor-Proyecto".</p>
Sección : "Buscar Profesor-Proyecto"	
Acción del Actor	Respuesta del Sistema
<p>1. Introduce uno de los parámetros por el cual desea realizar la búsqueda:</p> <ul style="list-style-type: none"> • Solapín • Nombre • Apellidos • Proyecto <p>2. El Moderador presiona el botón "Filtrar".</p>	<p>3. El sistema muestra un listado según el parámetro seleccionado por el Moderador.</p> <p>4. Finaliza el CU.</p>

Flujo Alternativo de la Sección : “Buscar Profesor-Proyecto”	
Acción del Actor	Respuesta del Sistema
	3.1 Si el sistema no encuentra profesores o proyectos con esas características muestra un listado vacío. Ir a la acción 1.
Sección : “Eliminar Profesor-Proyecto”	
Acción del Actor	Respuesta del Sistema
1. El Moderador escoge el profesor con el proyecto correspondiente que desea eliminar. 3. El Moderador presiona el botón “Aceptar”.	2. El sistema muestra un mensaje de confirmación. 4. El sistema elimina la relación del profesor con el proyecto correspondiente. 5. Muestra un mensaje y actualiza los datos. 6. Finaliza el CU.
Flujo Alternativo de la Sección : “Eliminar Profesor-Proyecto”	
Acción del Actor	Respuesta del Sistema
3.1 El Moderador presiona el botón “Cancelar”. 3.2 Finaliza el CU.	
Prioridad	Secundario.

Tabla 11. Descripción ampliada del caso de uso Asignar Estudiante-Proyecto.

Caso de Uso

CU-34	Asignar Estudiante-Proyecto.	
Actores	Moderador (inicia).	
Resumen	El caso de uso inicia cuando el Moderador escoge Asignar Estudiante-Proyecto. El sistema muestra una interfaz con los estudiantes que no tienen proyecto asignado, dando la posibilidad de buscar estudiantes para asignarle el proyecto. El caso de uso finaliza cuando se emite el resultado de la opción solicitada.	
Referencias	R91	
Precondiciones	El Moderador se encuentra autenticado.	
Poscondiciones	Es asignado el proyecto al estudiante.	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Sistema	
<p>1. El Moderador desea asignar proyecto a estudiante.</p> <p>3. El Moderador desea realizar una de las siguientes opciones:</p> <ul style="list-style-type: none"> • Buscar estudiantes definiendo criterio. • Seleccionar proyecto. 	<p>2. El sistema muestra una interfaz con todos los estudiantes que no tienen proyecto asignado y opciones para filtrar y/o seleccionar el proyecto.</p> <p>4. El sistema en dependencia de la opción que solicita realizar el Moderador, ejecuta las siguientes acciones:</p> <p>a) Si decide buscar estudiantes. Ver sección: "Buscar Estudiantes".</p> <p>b) Si decide seleccionar proyecto. Ver sección: "Seleccionar Proyecto".</p>	
Sección : "Buscar Estudiantes"		
Acción del Actor	Respuesta del Sistema	

<p>1. Introduce uno de los parámetros por el cual desea realizar la búsqueda:</p> <ul style="list-style-type: none"> • Solapín • Nombre • Apellidos <p>2. El Moderador presiona el botón “Filtrar”.</p>	<p>3. El sistema muestra un listado según el parámetro seleccionado por el Moderador.</p> <p>4. Finaliza el CU.</p>
Flujo Alternativo de la Sección : “Buscar Estudiantes”	
Acción del Actor	Respuesta del Sistema
	<p>3.1 Si el sistema no encuentra estudiantes con esas características muestra un listado vacío. Ir a la acción1.</p>
Sección : “Seleccionar Proyecto”	
Acción del Actor	Respuesta del Sistema
<p>1. El Moderador escoge el estudiante que desea asignarle el proyecto y presiona la opción Selecc.Proyecto.</p> <p>3. El Moderador desea realizar una de las siguientes opciones:</p> <ul style="list-style-type: none"> • Buscar proyectos definiendo criterio. • Asignar proyecto. 	<p>2. El sistema muestra una interfaz con todos los proyectos posibles a ser asignados y opciones para filtrar y/o asignar proyecto.</p> <p>4. El sistema, en dependencia de la opción que solicita realizar el Moderador, ejecuta las siguientes acciones:</p> <p>a) Si decide buscar proyectos. Ver sección: “Buscar Proyectos”.</p> <p>b) Si decide asignar proyecto. Ver sección: “Asignar Proyecto”.</p>

Sección : “Buscar Proyectos”	
Acción del Actor	Respuesta del Sistema
<p>1. Introduce uno de los parámetros por el cual desea realizar la búsqueda:</p> <ul style="list-style-type: none"> • Nombre • Fechas <p>2. El Moderador presiona el botón “Filtrar”.</p>	<p>3. El sistema muestra un listado según el parámetro seleccionado por el Moderador.</p> <p>4. Finaliza el CU.</p>
Flujo Alternativo de la Sección : “Buscar Proyectos”	
Acción del Actor	Respuesta del Sistema
	<p>3.1 Si el sistema no encuentra proyectos con esas características muestra un listado vacío. Ir a la acción1.</p>
Sección : “Asignar Proyecto”	
Acción del Actor	Respuesta del Sistema
<p>1. El Moderador escoge el proyecto que desea asignarle al estudiante y presiona la opción Asignar Proyecto.</p> <p>3. El Moderador presiona el botón “Aceptar.”</p>	<p>2. El sistema muestra un mensaje de confirmación.</p> <p>4. El sistema le asigna el proyecto al estudiante.</p> <p>5. El sistema emite un mensaje y muestra el listado actualizado de los estudiantes sin proyecto asignado.</p>

	6. Finaliza el CU.
Flujo Alternativo de la Sección : “Asignar Proyecto”	
Acción del Actor	Respuesta del Sistema
3.1 El Moderador presiona el botón “Cancelar.” 3.2 Finaliza el CU.	
Prioridad	Crítico.

Tabla 12. Descripción ampliada del caso de uso Eliminar Estudiante-Proyecto.

Caso de Uso	
CU-35	Eliminar Estudiante-Proyecto.
Actores	Moderador (inicia).
Resumen	El caso de uso inicia cuando el Moderador escoge Eliminar Estudiante-Proyecto. El sistema muestra la interfaz con los estudiantes y su proyecto correspondiente, dando la posibilidad de buscar para liberar a un estudiante de su proyecto correspondiente. El caso de uso finaliza cuando se emite el resultado de la opción solicitada.
Referencias	R92
Precondiciones	El Moderador se encuentra autenticado.
Poscondiciones	Es liberado un estudiante de su proyecto correspondiente.
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El Moderador desea liberar a un	2. El sistema muestra una interfaz con todos los

<p>estudiante de su proyecto correspondiente.</p> <p>3. El Moderador desea realizar una de las siguientes opciones:</p> <ul style="list-style-type: none"> • Buscar estudiante-proyecto definiendo criterio. • Liberar estudiante-proyecto específico. 	<p>estudiantes con su proyecto asignado y opciones para filtrar y/o liberar.</p> <p>4. El sistema, en dependencia de la opción que solicita realizar el Moderador, ejecuta las siguientes acciones:</p> <p>a) Si decide buscar estudiante-proyecto. Ver sección: “Buscar Estudiante-Proyecto”.</p> <p>b) Si decide liberar estudiante-proyecto. Ver sección: “Liberar Estudiante-Proyecto”.</p>
Sección : “Buscar Estudiante-Proyecto”	
Acción del Actor	Respuesta del Sistema
<p>1. Introduce uno de los parámetros por el cual desea realizar la búsqueda:</p> <ul style="list-style-type: none"> • Solapín • Nombre • Apellidos • Proyecto <p>2. El Moderador presiona el botón “Filtrar”.</p>	<p>3. El sistema muestra un listado según el parámetro seleccionado por el Moderador.</p> <p>4. Finaliza el CU.</p>
Flujo Alternativo de la Sección : “ Buscar Estudiante-Proyecto”	
Acción del Actor	Respuesta del Sistema
	<p>3.1 Si el sistema no encuentra estudiantes o proyectos con esas características muestra un listado vacío. Ir a la acción 1.</p>

Sección : “ Liberar Estudiante-Proyecto”	
Acción del Actor	Respuesta del Sistema
<p>1. El Moderador escoge el estudiante con el proyecto correspondiente que desea liberar.</p> <p>3. El Moderador presiona el botón “Aceptar”.</p>	<p>2. El sistema muestra un mensaje de confirmación.</p> <p>4. El sistema libera al estudiante de su proyecto correspondiente.</p> <p>5. El sistema emite un mensaje y muestra los datos actualizados.</p> <p>6. Finaliza el CU.</p>
Flujo Alternativo de la Sección : “ Liberar Estudiante-Proyecto”	
Acción del Actor	Respuesta del Sistema
<p>3.1 El Moderador presiona el botón “Cancelar”.</p> <p>3.2 Finaliza el CU.</p>	
Prioridad	Secundario.

GLOSARIO

A

ADO (ActiveX Data Objects): Mecanismo que usan los programas de computadoras para comunicarse con las bases de datos, darles órdenes y obtener resultados de ellas.

API (Application Programming Interface): Una Interfaz de Programación de Aplicaciones no es más que una serie de servicios o funciones que el Sistema Operativo ofrece al programador.

B

Backup: Respaldo o copia de seguridad que se realiza a cualquier aplicación, base de datos y otros.

Bluefish: Es un editor para programadores y diseñadores Web, publicado bajo licencia GPL y compatible con sistemas operativos POSIX (Portable Operating System Interface), lo que incluye Linux, FreeBSD y MacOS X. Aún soportando numerosos lenguajes, está enfocado en la edición de sitios Web dinámicos.

BSD: Es la licencia de software otorgada principalmente para los sistemas BSD (Berkeley Software Distribution). Pertenece al grupo de licencias de software Libre. La licencia BSD al contrario que la GPL permite el uso del código fuente en software no libre.

C

CCS (Cascading Sheets of Style): Hojas de estilo en cascada, archivo en el que se definen las características de diseño de una página Web.

CMS (Content Management System): Sistema que facilita la gestión de contenidos en todos sus aspectos: creación, mantenimiento, publicación y presentación. También se conoce como Web Content Management (WCM) sistema de gestión de contenido de Webs.

D

Disco duro: Dispositivo que permite el almacenamiento y recuperación de grandes cantidades de información. Los discos duros forman el principal elemento de la memoria secundaria de un ordenador, llamada así en oposición a la memoria principal o memoria RAM.

F

Flash: Software de Macromedia para crear pequeñas animaciones vectoriales reproducidas en la Web.

G

GNU: Acrónimo recursivo que significa GNU No es Unix (GNU is Not Unix), proyecto nacido en 1984 para desarrollar un sistema operativo similar a UNIX, pero bajo el concepto de software libre.

GPL (General Public License): Licencia que permite el uso y modificación del código para desarrollar software libre, pero no propietario.

H

Herramientas CASE (Computer Aided Software Engineering): Ingeniería de Software Asistida por Ordenador) son diversas aplicaciones informáticas destinadas a aumentar la productividad en el desarrollo de software reduciendo el coste de las mismas en términos de tiempo y de dinero.

HTML (HyperText Markup Language): Lenguaje basado en marcas que indican las características del texto, utilizado para definir documentos de hipertexto en webs.

HTTP (HyperText Transfer Protocol): Protocolo de transferencia de hipertexto, es un protocolo del nivel de aplicación usado para la transferencia de información entre sistemas de forma clara y rápida.

IDE (Integrated Development Environment): Entorno de desarrollo integrado, es un programa compuesto por un conjunto de herramientas para un programador.

K

KParts: Es el nombre del componente framework para el entorno de escritorio KDE.

KDE: Es un entorno de escritorio para GNU/Linux y otros sistemas derivados de Unix. Incluye muchas aplicaciones integradas entre sí (entre las más populares Konqueror), además un manejador de ventanas. Utiliza las librerías gráficas Qt para mostrar los elementos de la interfaz.

L

LAN (Local Area Network): Una red de área local o red local, es la interconexión de varios ordenadores y periféricos. Su extensión está limitada físicamente a un edificio o a un entorno de pocos kilómetros.

LDAP (Lightweight Directory Access Protocol): Protocolo de Acceso Ligero a Directorios es un protocolo de tipo cliente-servidor para acceder a un servicio de directorio.

M

Microprocesador: Micro o "unidad central de procesamiento", CPU, es un chip que sirve como cerebro de la computadora. En la actualidad en el interior de este componente electrónico existen millones de transistores integrados.

N

.NET: Es un proyecto de Microsoft para crear una nueva plataforma de desarrollo de software con énfasis en transparencia de redes, con independencia de plataforma de hardware y que permita un rápido desarrollo de aplicaciones.

O

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

P

PHP (PHP Hypertext Preprocessor): Lenguaje de programación para el desarrollo de webs dinámicas, con sintaxis parecida a la que originalmente se conocía como Personal Home Page Tools, herramientas para páginas personales (en Internet).

Plugins: Pequeños archivos que añaden mayores funcionalidades y características a tu blog.

Q

Qt: Es una biblioteca multiplataforma para desarrollar interfaces gráficas de usuario.

R

RAM (Random Access Memory): Memoria de Acceso Aleatorio, es donde el computador guarda los datos que está utilizando en el momento presente. El almacenamiento es considerado temporal porque los datos y programas permanecen en ella mientras que la computadora se encuentre encendida o no sea reiniciada.

RSS (Really Simple Syndication): Es parte da familia de formatos XML, desarrollado específicamente para los sitios Web que se actualizan con frecuencia, a través del RSS se puede compartir información y usarla en otros sitios Web o programas. A esto se conoce como sindicación o redifusión.

S

Skin: Imagen, conjunto de imágenes o archivo que permite cambiar la apariencia de un programa.

T

TCP/IP: Referencia a los dos protocolos más importantes que la componen: Protocolo de Control de Transmisión (TCP) y Protocolo de Internet (IP), que fueron los dos primeros en definirse, y que son los más utilizados de la familia.

Template: Una plantilla es una forma de dispositivo que proporciona una separación entre la forma o estructura y el contenido.

TIC: Tecnologías de la Información y la Comunicación, se encargan del estudio, desarrollo, implementación, almacenamiento y distribución de la información mediante la utilización de hardware y software como medio de sistema informático.

Triggers: O disparador en una base de datos, es un evento que se ejecuta cuando se cumple una condición establecida al realizar una operación de inserción (INSERT), actualización (UPDATE) o borrado (DELETE).

U

USB (Universal Serial Bus): El bus universal en serie, es un puerto que sirve para conectar periféricos a una computadora.

W

WAN (Wide Area Network): Red de Área Amplia, es un tipo de red de computadoras capaz de cubrir distancias desde unos 100 Km. hasta unos 1000 Km., brinda el servicio a un país o un continente.

Web: Es un sistema para presentar información en Internet basado en hipertexto. Cuando se utiliza en masculino (el Web, un Web) se refiere a un sitio Web entero, en cambio si se utiliza en femenino (la Web, una Web) se refiere a una página Web concreta dentro del sitio Web.

Webmaster: Es un término generalmente usado para referirse a las personas responsables de un sitio Web específico.

WYSIWYG: Los editores HTML que se denominan habitualmente WYSIWYG (What You See Is What You Get), la razón es debido a cuando se trabaja con ellos, lo que se observa que se crea con el editor es lo que se obtiene luego cuando se graba la página.

WWW (World Wide Web): O simplemente Web, es el universo de información accesible a través de Internet, una fuente inagotable del conocimiento humano.

X

XML (Extensible Markup Language): Lenguaje de marcas extensible, es un metalenguaje extensible de etiquetas desarrollado por el World Wide Web Consortium.

Z

Zope: Es un servidor de aplicaciones Web escrito en el lenguaje de programación Python. Puede ser manejado casi totalmente mediante el uso de una interfaz de usuario basada en páginas Web.