



Universidad de las Ciencias Informáticas.

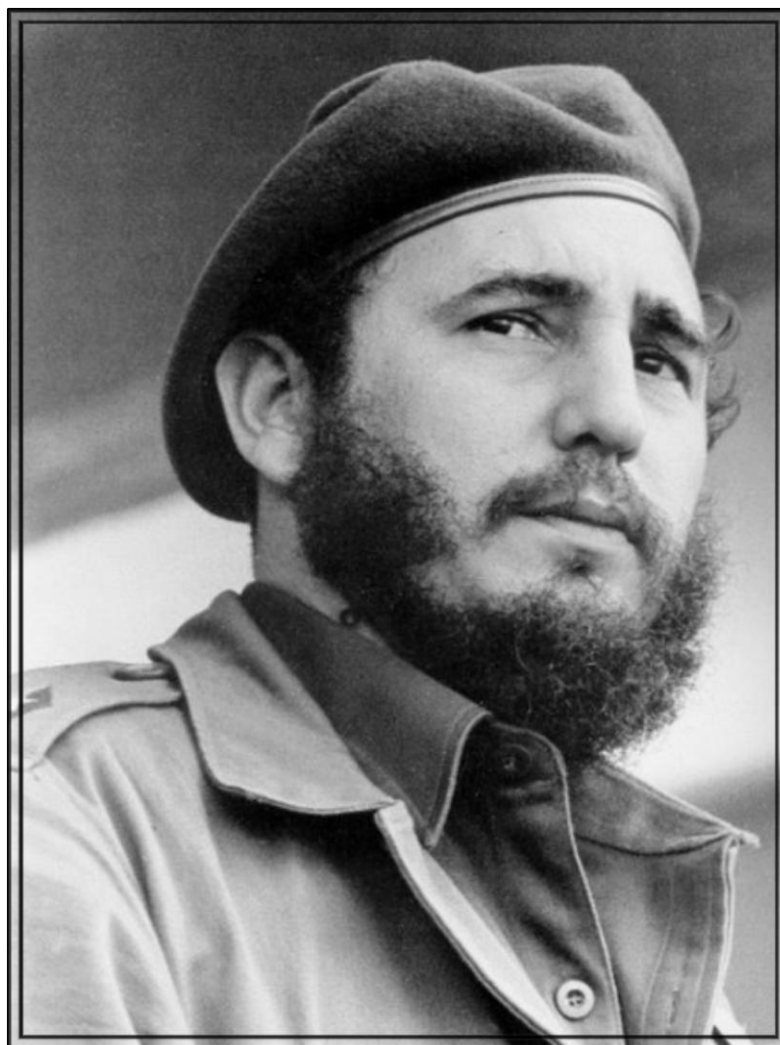
Portal Web de la Unión Nacional de Juristas de Cuba.

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

Autores: Arturo Cedeño Carbonell
Eusebio A. García Más

Tutor: Ing. Abdiel Matos Nieto.

Ciudad de la Habana, 2011



(...) “Nuestras funciones son bastante diferentes, sin embargo, aquí nos une un concepto somos hombres de derecho, hemos estudiado el derecho y nos hemos dedicados al derecho.” (...)

Fidel Castro Ruz

Discurso pronunciado en el acto de Celebración del Día del Jurista, en el Hotel Habana-Hilton, 8 de junio de 1959.

Declaración de Autoría

Declaro que soy el único autor de este trabajo y autorizo a la Facultad 3 de la Universidad de las Ciencias Informáticas a hacer uso del mismo en su beneficio.

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

Eusebio Andrés García Mas

Arturo Cedeño Carbonell

Firma del Autor

Firma del Autor

Ing. Abdiel Matos Nieto

Firma del Tutor

Datos de Contacto

Arturo Cedeño Carbonell

Email: acarbonell@estudiantes.uci.cu

Dirección Particular: Calle 5ta, # 9, e/ Ave. Victoriano Garzón y J.C.Z. Escario, Reparto Santa Bárbara, Santiago de Cuba, Santiago de Cuba.

Eusebio Andrés García Más

Email: eagarcia@estudiantes.uci.cu

Teléfono: 255-2208

Dirección Particular: Calle 318 Edificio 14 apto A, e/ 167A y 167B Valle Grande, La Lisa, La Habana.

Ing. Abdiel Matos Nieto

Graduado de Ingeniero en Ciencias Informáticas por la Universidad de las Ciencias Informáticas en el año 2009. Actualmente se encuentra laborando en la Universidad de las Ciencias Informáticas como profesor con la categoría docente de Instructor Recién Graduado.

Email: anieto@uci.cu

Teléfono: 053-796-677

Dirección Particular:

Resumen

El Portal de la Unión Nacional de Juristas de Cuba (UNJC) es la herramienta web que se encarga de publicar información a los usuarios de dicha organización. La UNJC es una organización de profesionales del Derecho, donde se agrupan actualmente más de catorce mil juristas.

El antiguo sitio web de la UNJC, era un sitio web con contenido estático, lo que deriva varios problemas como por ejemplo, poca interacción entre sus usuarios y con la UNJC, dificultad para hacer cambios en el sitio y un diseño pobre. Basado en estos problemas, se desarrolló un portal que les brinda a los usuarios de la UNJC muchos más servicios y comodidades, eliminando las deficiencias encontradas. Además, en Cuba, la red no es lo suficientemente rápida, por lo que el portal tiene en cuenta las particularidades de dicha red y brinda servicios de calidad y muy interactivos bajo los términos de la Web 2.0, sin ralentizar la carga del mismo, logrando así el acceso desde cualquier parte del país.

Palabras Clave: Unión Nacional de Juristas de Cuba, Portal Web, Web 2.0.

ÍNDICE

ÍNDICE DE ILUSTRACIONES	VI
INTRODUCCIÓN	1
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA.....	5
INTRODUCCIÓN	5
1.1 LA WEB 2.0.....	5
1.1.1 Redes Sociales	6
1.1.2 Blog o Bitácora.....	6
1.1.3 Servicios Web.....	6
1.1.4 Wikis.....	7
1.1.5 Mashups.....	7
1.1.6 Aplicaciones Web.....	7
1.1.7 Portales Web.....	8
1.2 LA WEB 2.0 EN CUBA	9
1.2.1 INFOMED	9
1.3 METODOLOGÍAS DE DESARROLLO	10
1.3.1 Programación Extrema	11
1.3.2 Proceso Unificado de Rational	13
1.3.3 Proceso Unificado Ágil	15
1.3.4 Guía Web	16
1.4 LENGUAJES DE PROGRAMACIÓN	18
1.4.1 PHP.....	18
1.4.2 XHTML.....	19
1.4.3 JavaScript.....	20
1.4.4 UML.....	21
1.5 SISTEMAS GESTORES DE CONTENIDOS	22
1.5.1 Drupal	23
1.5.2 Joomla!.....	24
1.6 SISTEMAS GESTORES DE BASE DE DATOS	25
1.6.1 MySQL.....	25

1.7	ENTORNO DE DESARROLLO INTEGRADO PARA APLICACIONES WEB.....	27
1.7.1	<i>Aptana Studio</i>	27
1.7.2	<i>Zend Studio</i>	28
1.8	HERRAMIENTAS DE INGENIERÍA DE SOFTWARE ASISTIDA POR COMPUTADORA.....	29
1.8.1	<i>Rational Rose</i>	29
1.8.2	<i>Visual Paradigm</i>	30
	CONCLUSIONES.....	31
CAPÍTULO 2: CARACTERÍSTICAS DEL PORTAL WEB		32
	INTRODUCCIÓN.....	32
2.1	DESCRIPCIÓN DEL PORTAL WEB.	32
2.2	LEVANTAMIENTO DE REQUISITOS.....	32
2.2.1	<i>Requisitos Funcionales</i>	33
2.2.1.1	Subsistema Sociedades Científicas:	33
2.2.1.2	Subsistema Juntas Directivas:.....	33
2.2.1.3	Subsistema Administración de Usuarios de la UNJC:.....	34
2.2.1.4	Subsistema Eventos de la UNJC:	34
2.2.1.5	Subsistema Servicios de la UNJC:.....	35
2.2.2	<i>Requisitos no Funcionales</i>	35
2.2.2.1	Usabilidad	35
2.2.2.2	Confiabilidad.....	37
2.2.2.3	Eficiencia.....	37
2.2.2.4	Soporte	38
2.2.2.5	Restricciones de Diseño	38
2.2.2.6	Requisitos para la documentación de usuarios en línea y ayuda del sistema.	38
2.2.2.7	Requisitos de Licencia.....	38
2.3	CASOS DE USO DEL SISTEMA.....	38
2.3.1	<i>Diagrama de Casos de Uso del Sistema</i>	40
2.3.2	<i>Actores del Sistema</i>	40
2.3.3	<i>Descripción de los Casos de Uso del Sistema</i>	41
2.3.3.1	Caso de Uso Gestionar Sociedades Científicas.	41
	CONCLUSIONES.....	47
CAPÍTULO 3: IMPLEMENTACIÓN Y PRUEBA DEL PORTAL WEB		48

INTRODUCCIÓN	48
3.1 EL CMS JOOMLA!	48
3.1.1 <i>Patrón N-Capas</i>	48
3.1.2 <i>Patrón Modelo-Vista-Controlador</i>	49
3.1.3 <i>Arquitectura del CMS Joomla!</i>	49
3.1.4 <i>Estructura de un Componente del CMS Joomla!</i>	51
3.2 IMPLEMENTACIÓN	51
3.2.1 <i>Componente Juntas Directivas</i>	51
3.2.2 <i>Componente Sociedades Científicas</i>	53
3.2.3 <i>Componente ChronoForms</i>	53
3.3 PRUEBAS	54
3.3.1 <i>Pruebas de caja blanca</i>	54
3.3.2 <i>Pruebas de caja negra</i>	54
3.3.3 <i>Pruebas a los componentes desarrollados</i>	55
3.3.3.1 Caso de Prueba Gestionar Juntas Directivas	55
3.3.3.1 Caso de Prueba Gestionar Sociedades Científicas	58
3.3.4 <i>Pruebas de Rendimiento</i>	62
CONCLUSIONES	63
CONCLUSIONES	64
RECOMENDACIONES	65
REFERENCIAS BIBLIOGRÁFICAS	66
APÉNDICES	70
GLOSARIO DE TÉRMINOS	70

ÍNDICE DE ILUSTRACIONES

Figure 1 Metodología XP	12
Figure 2 Metodología Proceso Unificado de Rational.....	14
Figure 3 Metodología Proceso Unificado Ágil.....	16
Figure 4 Diagrama de Casos de Uso del Sistema.....	40
Figure 5 Patrón de Arquitectura N-Capas	49
Figure 6 Arquitectura de Joomla!	50
Figure 7 Subsistema Juntas Directivas	52
Figure 8 Diseño del Subsistema Juntas Directivas.....	52
Figure 9 Ejemplo de Código implementado para el Componente Chronoforms	54
Figure 10 SC Crear Junta Directiva	56
Figure 11 SC Modificar Junta Directiva	57
Figure 12 SC Eliminar Junta Directiva.....	58
Figure 13 SC Buscar Junta Directiva	58
Figure 14 SC Crear Sociedad Científica	59
Figure 15 SC Modificar Sociedad Científica	60
Figure 16 SC Eliminar Sociedad Científica.....	61
Figure 17 SC Buscar Sociedad Científica	61

INTRODUCCIÓN

Desde el inicio de la World Wide Web (W3 o comúnmente llamada Web), las empresas vieron su surgimiento como una gran oportunidad para hacer negocios, pues, con la creación de ésta, se elimina la limitación de sólo negociar en un área geográfica en específico, pues se puede negociar online desde cualquier parte del mundo, hacia cualquier parte del mundo. Al principio, la Web no es lo que es hoy, las páginas web antiguas eran páginas estáticas y los usuarios solo podían ver su contenido de forma pasiva (visualización pasiva). A esa forma antigua de publicación de sitios web se le llamó Web 1.0.

Con el tiempo, los sitios web fueron evolucionando y se les dio mayor protagonismo a los usuarios, donde pueden crear y editar contenidos online, las empresas interactuar con sus usuarios, se pueden personalizar las páginas y otras características, lo que es hoy, la Web 2.0. Actualmente se están trazando las pautas para la implementación de una web inteligente, la cual se le llama web semántica o simplemente Web 3.0.

Dado que las empresas necesitaban mostrar sus productos o servicios a todos sus clientes potenciales en el mundo, se crean los portales web, que su función principal no es más que la de servir de puerta de entrada única para ofrecer al usuario, de forma fácil e integrada, el acceso a los recursos y servicios que brinda, incluyendo buscadores, enlaces, foros, entre otros. De esta manera, queda publicada en un sitio web, la imagen de la empresa, por lo que es muy necesario que el portal denote el profesionalismo y seriedad de la entidad en cuestión, pues el portal es su primer y principal recurso para lograr una buena imagen en los clientes. Ejemplos de portales son el de Microsoft Corporation (<http://www.microsoft.com>) y el de IBM (<http://www.ibm.com>).

Al pasar el tiempo, no solo las empresas publicaron sus portales, sino que muchas asociaciones gubernamentales, no gubernamentales y personas lanzaron sus propios sitios webs, cada uno con sus características propias. Como ejemplo de asociaciones no gubernamentales se encuentra el portal de la Comisión Internacional de Juristas (en lo adelante **ICJ**, por sus siglas en inglés **International Commission of Jurists**).

En Cuba la situación no fue muy distinta, el gobierno cubano hizo grandes esfuerzos para poder incluir al país en este proceso de ofrecer los productos y servicios de empresas y organizaciones gubernamentales y no gubernamentales en la Web. La Unión Nacional de Juristas de Cuba (en lo adelante **UNJC**), como

asociación no gubernamental, dio su primer paso con la creación de su sitio web bajo la dirección de <http://www.unjc.co.cu>.

La UNJC es una organización de profesionales del Derecho, con status no gubernamental y personalidad jurídica propia, constituida el 8 de junio de 1977, sin ánimo de lucro, autofinanciada, y con un perfil científico-profesional. En la UNJC se agrupan actualmente catorce mil juristas cubanos asociados voluntariamente, los que se desempeñan como profesores universitarios, investigadores, abogados, jueces, fiscales, notarios, consultores, asesores y otros. (Unión Nacional de Juristas de Cuba, 1977)

Actualmente, el sitio web de la UNJC tiene las características de poseer contenido estático, poca actualización, nula interoperabilidad y colaboración entre los usuarios, ausencia de metadatos, entre otras deficiencias, en fin, un sitio con las características de la Web 1.0. Como el sitio web existente tiene estos problemas, entonces trae sus consecuencias, dado que el mantenimiento, actualización de la información y otras tareas comunes se convierte en un proceso tedioso y extremadamente costoso en tiempo y recursos. Otros problemas presentes son, por ejemplo, que no se obtienen muchas visitas de los usuarios por la poca actualización de contenidos, debido a la ausencia de metadatos y el uso deficiente o inexistente de nuevas tecnologías usadas por los motores de búsqueda, no se indexan correctamente las páginas en los buscadores más usados y no se obtienen datos importantes como la cantidad de veces que es visto un artículo determinado, las secciones del sitio más visitadas y otros datos acerca del uso que le dan los usuarios al sitio web.

Para darle solución a la problemática planteada se define el siguiente **problema a resolver**: ¿Cómo lograr contenido dinámico, contribuir al aumento de la interoperabilidad y colaboración en línea, presencia de metadatos y parámetros de usabilidad en el sitio web de la UNJC?

El **objeto de estudio** de esta investigación lo constituye el proceso de desarrollo de software, definiendo como el **campo de acción** el proceso de desarrollo de portales web.

El **objetivo general** de la investigación es desarrollar el Portal Web de la UNJC para lograr contenido dinámico, contribuir al aumento de la interoperabilidad y colaboración en línea, presencia de metadatos y parámetros de usabilidad.

Como **idea a defender** se plantea que: si se desarrolla el nuevo Portal Web de la UNJC, se lograría mayor facilidad para compartir información, contribuiría a lograr mayor interoperabilidad y colaboración en línea, además de permitir a sus usuarios una visualización activa de la información que se publica.

Para dar cumplimiento al objetivo general se derivan los siguientes **objetivos específicos**:

- Obtener requisitos funcionales y no funcionales del Portal Web para conocer cuáles deben ser las funcionalidades y cualidades a desarrollar mediante la realización de entrevistas con los usuarios de la UNJC.
- Diseñar las extensiones a implementar en el Portal Web para obtener una comprensión más detallada y encontrar una forma que soporte todos los requisitos.
- Implementar las extensiones para el Portal Web para dar solución a los requisitos identificados.
- Realizar pruebas al Portal Web para verificar su correcto funcionamiento a través de pruebas funcionales y no funcionales.

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

- Realización de un estado del arte donde se tengan en cuenta las características de la Web 2.0, las metodologías de desarrollo de software y de aplicaciones Web, los lenguajes de programación web libres, los sistemas gestores de contenido, los sistemas gestores de bases de datos y las herramientas a utilizar para el desarrollo del Portal.
- Licitación y especificación de los requisitos funcionales y no funcionales del Portal.
- Realización del diseño de las extensiones para el Portal.
- Implementación de las extensiones para el Portal.
- Realización de pruebas funcionales y no funcionales del Portal.

Con el objetivo de alcanzar la solución deseada y de facilitar el desarrollo de la investigación se utilizaron los siguientes métodos de investigación:

Métodos teóricos:

- El método Analítico - Sintético se utilizó con el fin de analizar libros, páginas web y otras bibliografías para buscar los elementos más importantes que se relacionan con esta investigación.
- El método Histórico-Lógico se utilizó con el objetivo de verificar teóricamente cómo ha evolucionado el tema tratado en esta investigación.

- El método Modelación se utilizó con el objetivo de reproducir la interacción de los objetos en la vida real mediante la creación de modelos.

Métodos empíricos:

- La Entrevista se utilizó con el objetivo de obtener información valiosa para el desarrollo de la investigación.

El presente trabajo estará conformado de la siguiente manera: introducción, tres capítulos, conclusiones, recomendaciones, referencias bibliográficas y apéndices. Los capítulos abordan los siguientes temas:

CAPÍTULO 1: Fundamentación teórica. Incluye un estado del arte de la Web 2.0 y sus características y aplicaciones web con funcionalidad similar al sistema a desarrollar. Se muestra un estudio de las Metodologías de Desarrollo de Software y de Aplicaciones Web, de los Lenguajes de Programación Web, los Entornos de Desarrollo Integrado, Sistemas Gestores de Contenidos, Sistemas Gestores de Bases de Datos y Herramientas de Ingeniería de Software Asistida por Computadora a utilizar en el desarrollo del Portal.

CAPÍTULO 2: Características del Portal Web. Define las principales características que debe cumplir el sistema a desarrollar en términos de requisitos funcionales y no funcionales. Se enumeran los requisitos funcionales y no funcionales del sistema, mediante Diagramas de Casos de Uso se describen las relaciones existentes de los actores que se involucran con el sistema y mediante la descripción textual de casos de uso se explica la propuesta de funcionamiento del sistema.

CAPÍTULO 3: Implementación y Prueba del Portal Web. Se describe el funcionamiento del CMS Joomla! en su versión 1.6, se hace referencia a los patrones arquitectónicos que se utilizan en Joomla! y en las extensiones desarrolladas, se implementan las funcionalidades que requiere el Portal y se describen las pruebas funcionales y no funcionales a las extensiones construidas y se dan a conocer los resultados de dichas pruebas.

CAPÍTULO 1: Fundamentación teórica

Introducción

El presente capítulo incluye un estado del arte de la Web 2.0 y sus características y aplicaciones web con funcionalidad similar al sistema a desarrollar. Se muestra un estudio de las Metodologías de Desarrollo de Software y de Aplicaciones Web, de los Lenguajes de Programación Web, los Entornos de Desarrollo Integrado, Sistemas Gestores de Contenidos, Sistemas Gestores de Bases de Datos y Herramientas de Ingeniería de Software Asistida por Computadora a utilizar en el desarrollo del Portal. El estudio se hará teniendo en cuenta las particularidades de la red cubana y las opciones de hospedaje web que posee la UNJC.

1.1 La Web 2.0

Para definir qué es Web 2.0, se toma en cuenta el concepto planteado por Tim O'Reilly, en una conferencia impartida en Octubre de 2004, en la ciudad de San Francisco, California; donde expresa que:

“El término Web 2.0 está comúnmente asociado con un fenómeno social, basado en la interacción que se logra a partir de diferentes aplicaciones en la Web, que facilitan el compartir información, la interoperabilidad, el diseño centrado en el usuario y la colaboración en Internet. Un sitio Web 2.0 permite a sus usuarios interactuar con otros usuarios o cambiar contenido del sitio web, en contraste a sitios web no-interactivos donde los usuarios se limitan a la visualización pasiva de información que se les proporciona” (¿Qué es Web 2.0?, 2004).

La red original, llamada Web 1.0, se basaba en páginas de contenido estático, creadas en el Lenguaje de Marcado de Hipertexto (en lo adelante **HTML**, por sus siglas en inglés **HyperText Markup Language**) que no eran actualizadas frecuentemente. El éxito de las páginas comerciales (*.com) dependía de lograr contenidos dinámicos usando algún CMS, los cuales creaban páginas HTML desde una base de datos. En aquellos momentos, obtener el mayor número de visitas y la estética visual de las páginas web se consideraban como factores importantes.

La Web 2.0 es una nueva forma de aprovechar la red, permite participación activa de los usuarios, dándole autonomía en la web, permitiéndole administrar sus propios contenidos, opinar sobre otros y enviar y recibir información con otros usuarios o instituciones. Actualmente, existen varios tipos de sistemas con características de la Web 2.0, los cuales se muestran a continuación.

1.1.1 Redes Sociales

Las redes sociales son estructuras sociales compuestas de grupos de personas, las cuales están conectadas por uno o varios tipos de relaciones, tales como amistad, parentesco, intereses comunes o que comparten conocimientos (Freeman, 2004). De todas las formas actuales de aplicaciones web, las redes sociales ocupan el número uno en cuanto al número de visitas y al número de usuarios registrados en esos sistemas. Como ejemplo de las redes sociales más exitosas del mundo se pueden mencionar Facebook, Twitter, MySpace, entre otras.

1.1.2 Blog o Bitácora

Un blog, o en español también llamado bitácora, es un sitio web que contiene anotaciones o historias (conocidas como posts), cada una con una dirección específica, suelen aparecer en orden cronológico inverso (las últimas aparecen en primer lugar), y que admiten ser agrupadas por categorías. Suelen ser de un solo autor aunque existen blog colectivos o corporativos y van acompañados de comentarios de los usuarios que visitan las páginas. Se caracterizan por ser actualizados constantemente y por un estilo no formal de escritura. (Orihuela, 2006)

Este término inglés blog o weblog proviene de las palabras web y log ('log' en inglés = diario). El término bitácora, proviene en referencia a los antiguos cuadernos de bitácora de los barcos. En cada artículo, los lectores pueden escribir sus comentarios y el autor darles respuesta, de forma que se establece una conversación en línea, aunque el autor puede decidir que ningún lector pueda opinar. El tema de cada blog es particular, existen de tipo periodístico, empresarial o corporativo, tecnológico, educativo, políticos, personales, etc.

1.1.3 Servicios Web

Un servicio web (en inglés, *Web Service*) es un conjunto de aplicaciones o de tecnologías con capacidad para intercambiar datos en la red. Se intercambian datos entre sí con el objetivo de ofrecer servicios, los cuales son publicados por el proveedor como procedimiento remoto y los usuarios consumen un servicio llamando a estos procedimientos a través de la Web (World Wide Web Consortium España, 2010). No importa en qué lenguaje de programación ni sobre qué plataforma estén el proveedor y el cliente, la interoperabilidad se consigue mediante la adopción de estándares abiertos, por lo que se logra el intercambio de datos.

1.1.4 Wikis

Un wiki (o una wiki) es un sitio web cuyas páginas pueden ser editadas por múltiples voluntarios a través del navegador web. Los usuarios pueden crear, modificar o borrar un mismo texto que comparten. Los textos o páginas wiki tienen títulos únicos. Es mucho más sencillo y fácil de usar que una base de datos. Una característica que define la tecnología wiki es la facilidad con que las páginas pueden ser creadas y actualizadas, dando una gran libertad al usuario, y por medio de una interfaz muy simple (Cunningham, 1995).

En general, no hace falta revisión por la parte administrativa del sitio para que los cambios en las páginas web sean aceptados. Las wikis actuales tienen un historial de cambios, como los sistemas concurrentes de versionado (**CVS**, por sus siglas en inglés **C**oncurrent **V**ersioning **S**ystem) que permite recuperar de manera muy fácil cualquier estado anterior y ver qué usuario hizo cada cambio, lo que logra mayor facilidad en el mantenimiento del sitio y el control de usuarios destructivos.

1.1.5 Mashups

Una aplicación de tipo Mash-up es un sitio web que accede a datos o servicios de terceros y los combina para crear una nueva aplicación (Mashup, sumando en la red, 2008).

Los mash-up usan uno o varios proveedores de contenidos, los cuales son los verdaderos dueños de los datos o los servicios, los que se combinan en el mash-up, dando lugar a una nueva aplicación. Estos datos o servicios se obtienen generalmente usando el formato de sindicación realmente simple (en lo adelante **RSS**, por sus siglas en inglés **R**eally **S**imple **S**yndication), servicios web u otros.

1.1.6 Aplicaciones Web

Constituyen aquellas aplicaciones que los usuarios pueden utilizar accediendo a un servidor web a través de Internet o de una intranet mediante un navegador. En otras palabras, es una aplicación de software cuya interfaz de usuario es accesible desde un cliente web o navegador (Introducción a las Aplicaciones Web, 2004).

Todas las aplicaciones web están formadas por una arquitectura cliente servidor, la cual es básicamente un entorno distribuido en la que un programa, el servidor, gestiona un recurso compartido concreto y hace determinadas funciones solo cuando las pide otro, el cliente, que es quien interactúa con el usuario (Campderrich Falgueras, 2003).

El cliente, el cual comúnmente es el navegador web, se comunica con el servidor mediante el Protocolo de Transferencia de Hipertexto (en lo adelante **HTTP**, por sus siglas en inglés **H**ypertext **T**ransfer **P**rotocol) usando el Protocolo de Control de Transmisión y Protocolo de Internet (en lo adelante **TCP/IP**, por sus siglas en inglés **T**ransmission **C**ontrol **P**rotocol / **I**nternet **P**rotocol).

Algunas de las ventajas conocidas son:

- Ahorra tiempo: No se necesita instalar ningún programa adicional para usarlas, sólo el navegador web.
- No hay problemas de compatibilidad: Si el navegador web cumple con los estándares, las páginas web se ven casi iguales, sin importar el tipo de navegador ni el sistema operativo que se utilice.
- Ocupa muy poco espacio en el disco duro del cliente (pocos kilobytes).
- Actualizaciones inmediatas: Como el cliente utiliza solamente el navegador web, todas las actualizaciones de software se realizan en el servidor, por lo que siempre se estará usando la última versión disponible.
- Consumo de recursos bajo: La mayor parte de la aplicación no se encuentra en la estación cliente, por lo que la mayor parte del trabajo se realiza en el servidor.

Ejemplo de los inconvenientes en las aplicaciones web son:

- La disponibilidad depende de un tercero, depende del servidor o de los Proveedores de Servicios de Internet (en lo adelante **ISP**, por sus siglas en inglés **I**nternet **S**ervice **P**rovider).
- Por lo general brindan menos opciones que las aplicaciones de escritorio.
- Son más lentas en ejecución que las aplicaciones de escritorio, debido a la transmisión de datos por la red, la cual puede ser lenta, estar congestionada, entre otros factores.

1.1.7 Portales Web

Un Portal Web o Portal de Internet es un sitio web cuya característica fundamental es la de reunir, de forma fácil e integrada, el acceso a una serie de recursos y de servicios relacionados a un mismo tema, como son: enlaces, buscadores, foros, documentos, aplicaciones, correo electrónico, etc. (Nafría, 2008)

Un sitio web, para obtener el calificativo de portal tiene que cumplir con tres elementos fundamentales:

- El contenido tiene que ser de actualidad, específicos o generalistas.

- Poseer una serie de contenidos relacionados al tema del portal, tales como buscadores, envío de mensajes, datos de contacto, correo electrónico, entre otros.
- Poseer diversidad de herramientas para permitir la personalización de las páginas de los usuarios, permitir interacción del usuario con el portal e interacción entre los usuarios. Pueden ser herramientas como foros, chat y otras.

En el mundo, existen muchas variedades de portales, los cuales son clasificados de forma muy variada. Con el análisis de varias bibliografías, se puede observar que se plantea la existencia de portales móviles, diagonales, corporativos, especializados, de voz, inalámbricos, entre otros. En donde se puede hallar algo común, es que todas plantean la existencia de los portales horizontales y los verticales.

Los portales horizontales, se dirigen a una gran cantidad de personas, tratando de llegar a la mayor cantidad de personas con la mayor cantidad de temas y productos. Como ejemplo en esta categoría están Yahoo! y AltaVista.

Los portales verticales, ofrecen servicios y contenido dentro de un tema específico, para un sector en específico, como puede ser un portal de música, empleo, etc. Ejemplo de este tipo de portales lo constituye el portal de Infomed, el portal de la ICJ, entre otros.

El Portal de la UNJC, al igual que el de la ICJ, es para ofrecer contenidos referidos a un tema específico, el tema jurídico, por lo que el Portal Web de la UNJC se clasifica como un portal vertical.

1.2 La Web 2.0 en Cuba

En Cuba, también se pueden observar sitios web con características de la Web 2.0, siendo el más importante para esta investigación el portal de INFOMED, por sus características parecidas al sistema a desarrollar y por ser una de las entidades pioneras en la web cubana. Las características de este portal se mencionan a continuación.

1.2.1 INFOMED

Infomed es la red de personas e instituciones que trabajan y colaboran para facilitar el acceso a la información y el conocimiento para mejorar la salud de los cubanos y de los pueblos del mundo, mediante el uso intensivo y creativo de las tecnologías de la información y la comunicación. (Centro Nacional de Información de Ciencias Médicas, 2009)

Infomed brinda servicios en el área de salud, dentro de los cuales se pueden observar:

- Libros: Ofrece un vínculo a la referencia de libros y obras biomédicas disponibles en Internet.
- Revistas: Ofrece la referencia a revistas biomédicas en formato electrónico.
- Documentos: Ofrece un enlace directo a los documentos normativos del Sistema Nacional de Salud.
- Medicamentos: Base de Datos con la información farmacológica de todos los medicamentos que se utilizan para la atención a la salud en Cuba, en correspondencia con el Cuadro Básico de Medicamentos del país.
- Cursos: Este servicio está concebido para facilitar la búsqueda y divulgación de cursos de salud, tanto nacionales como internacionales, virtuales y presenciales.
- Eventos: Brinda información acerca de eventos como fórums, congresos, entre otros.
- Estadísticas: Acceso a la base de datos del Anuario Estadístico de Salud Cuba con todos los datos disponibles desde 1995 así como a los Indicadores Básicos de Salud.
- Al día: Servicio diario de actualización que ofrece una selección de las noticias internacionales más recientes en temas de salud y las ciencias médicas.
- Directorios: Posibilita el acceso a los directorios de instituciones de salud de Cuba y el de Unidades de la Red de Información de Ciencias Médicas.
- Correo: Brindan servicio de correo electrónico para los trabajadores de la salud en Cuba.
- Buscador: Brindan un servicio de buscador en el área de salud con el buscador llamado LIS.
- Hospedaje: Ofrece servicio de hospedaje web.
- Anuncios: Cualquier noticia relacionada con el soporte de los productos y servicios de Infomed.
- Sitios de Salud: Esta página permite acceder a los sitios web de las Instituciones de Salud de Cuba de relevancia nacional e internacional, incluyendo Hospitales, Institutos de Investigación, Centros de Atención Médica Especializada, entre otros.
- Cultura: Muestra galerías de imágenes museos entre otras cosas acerca de la cultura nacional.
- Tiempo: Brinda noticias acerca del tiempo.

1.3 Metodologías de Desarrollo

Antes de definir lo que es una metodología de desarrollo de software, es necesario preguntarse qué es un proceso de desarrollo de software, que no es más que un proceso el cual define quién está haciendo qué, cuándo y cómo alcanzar un determinado objetivo. (Jacobson, et al., 2000)

Una metodología de desarrollo de software es un conjunto de pasos y procedimientos que deben seguirse para desarrollar software. Una metodología define cómo dividir un proyecto en etapas, qué tareas se llevan a cabo en cada etapa, qué restricciones deben aplicarse, qué técnicas y herramientas se emplean y cómo se controla y gestiona un proyecto (Hernando, 2009).

Constituye un medio de estandarización que cubre por completo el proceso de desarrollo de software. Posibilita verificar trabajos realizados y corrección de los errores detectados y además provee un lenguaje común entre los analistas, programadores, clientes y usuarios.

Las metodologías de desarrollo de software se dividen en dos grandes grupos, las pesadas o tradicionales (Proceso Unificado de Rational) y las ágiles (Programación Extrema, Proceso Unificado Ágil). Las metodologías tradicionales se basan en la idea de que el éxito del producto se puede lograr si se tiene todo correctamente documentado, mientras que, las ágiles defienden la idea de que el proceso de desarrollo del software, se centra en el software como tal y no en la documentación alrededor de este, por lo que le da mayor importancia a la programación que a la documentación, aunque no la obvia por completo, sino que se toma en cuenta sólo la documentación necesaria y de forma muy sencilla.

Una metodología especifica completamente el desarrollo de un sistema informático, pero es necesario tener en cuenta que no es lo mismo hacer uso de un modelo para la creación de un sitio web que para el desarrollo de una aplicación de escritorio, por lo que existe una gran diversidad de metodologías de desarrollo. Debido a la gran variedad de metodologías que existen, es necesario hacer una buena elección, en dependencia de las características del sistema a desarrollar, el personal y la organización, entre otros factores. Se decide hacer un estudio de las metodologías de desarrollo de software más usadas como son el Proceso Unificado de Rational, Programación Extrema y el Proceso Unificado Ágil. Además, como se trata de un sistema web, se estudia la metodología de desarrollo de aplicaciones web Guía Web.

1.3.1 Programación Extrema

La Programación Extrema (en lo adelante **XP**, por sus siglas en inglés **eXtreme Programming**), es una metodología ágil, nacida en el verano de 1996, de la mano de Kent Beck. La misma se encuentra centrada en potenciar las relaciones interpersonales como clave para el éxito en el desarrollo de software, promoviendo el trabajo en equipo, preocupándose por el aprendizaje de los desarrolladores y propiciando

un buen clima de trabajo. Le da prioridad a las tareas que dan un resultado directo y que reducen la burocracia que hay alrededor de la programación. (Beck, 1999)

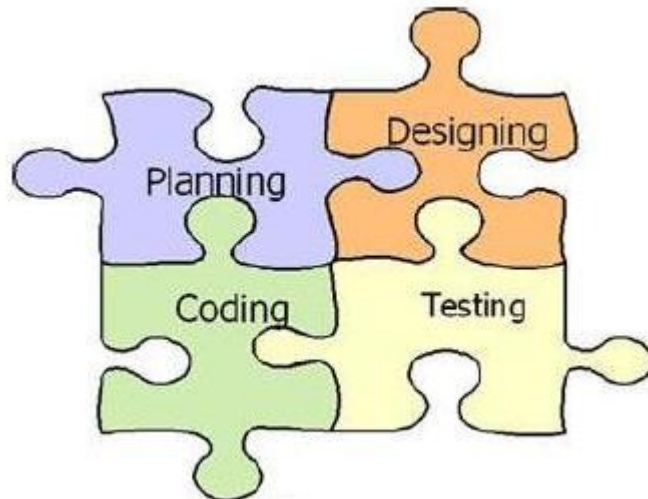


Figure 1 Metodología XP

XP, se basa en la realimentación continua entre el cliente y el equipo de desarrollo, comunicación fluida entre todos los participantes, simplicidad en las soluciones implementadas y responsabilidad a la hora de afrontar los cambios. Es especialmente adecuada para proyectos con requisitos imprecisos y muy cambiantes, y donde existe un alto riesgo técnico. (Beck, 1999)

La metodología XP se caracteriza por:

- Pruebas Unitarias: Estas pruebas se ejecutan en un entorno de desarrollo basado en pruebas, donde éstas se implementan antes de programar la funcionalidad deseada, así cuando se construye la funcionalidad, se programa para pasar la prueba ya escrita, se logra adelantar en tiempo y se pueden predecir los posibles errores que puedan ocurrir.
- Refabricación: Se basan en la reutilización de código por medio de la utilización de patrones de diseño y modelos, logrando que la aplicación sea más flexible al cambio.

- Programación en pares: Consiste en que los desarrolladores trabajen por parejas en una misma estación de trabajo, donde cada miembro realiza una tarea que el otro no esté haciendo en ese momento.

XP promueve que se empiece a desarrollar en pequeño y se añada funcionalidad con retroalimentación continua, que el manejo de los cambios sea parte sustantiva del proceso, no introducir funcionalidades antes de que sean necesarias y que el cliente o el usuario sea miembro del equipo de desarrollo.

El sistema a desarrollar cuenta con requisitos bien definidos y estables, necesita ser documentado de forma eficiente para los usuarios finales y posterior uso y el cliente no forma parte del equipo de trabajo, por situaciones ajenas al desarrollo, motivo por el cual XP no es la metodología más óptima para el desarrollo del Portal.

1.3.2 Proceso Unificado de Rational

El Proceso Unificado de Rational (en lo adelante **RUP**, por sus siglas en inglés **Rational Unified Process**) constituyó el resultado de tres décadas de desarrollo y trabajo práctico. Transitó desde el Proceso Objectory con su primera publicación en 1987 hasta el Proceso Objectory del Rational en 1997, para luego en 1998, tomar el nombre de RUP. Es un proceso de desarrollo de software que ofrece un conjunto de actividades para transformar los requerimientos de un usuario en un sistema informático.

RUP es un conjunto de metodologías adaptables al contexto y necesidades de cada organización, que en conjunto con el Lenguaje Unificado de Modelado (en lo adelante **UML**, por sus siglas en inglés **Unified Modeling Language**), constituye la metodología estándar más utilizada para el diseño, implementación y documentación de sistemas informáticos.

Las características fundamentales de RUP es que propone un desarrollo iterativo e incremental, centrado en la arquitectura y dirigido por casos de uso. Las mejores prácticas que RUP propone son:

- Desarrollo iterativo e incremental del software: Durante el ciclo de vida del desarrollo de un software, es mucho más fácil y práctico dividirlo en sub problemas más pequeños y manejables o mini proyectos, donde cada mini proyecto es llamado iteración, que resulta en un incremento. Las

iteraciones referencian los distintos pasos seguidos en los flujos de trabajo y los incrementos al crecimiento del producto final. (Jacobson, et al., 2000)

- Proceso centrado en la arquitectura: El objetivo de la arquitectura de software es muy parecido al objetivo de la arquitectura de los edificios, que permite al constructor tener una imagen completa, desde diferentes puntos de vista, antes de comenzar la construcción. La arquitectura de software tiene en cuenta los aspectos estáticos y dinámicos más significativos, las necesidades de la empresa y clientes reflejadas en casos de uso y muchos otros factores como la plataforma en que funciona (Software, Hardware, etc...), componentes reutilizables, requisitos no funcionales, etc. (Jacobson, et al., 2000)
- Proceso dirigido por casos de uso: Todo sistema de software se hace para brindar un servicio a un grupo de usuarios determinado, por usuario se entiende tanto a personas como a otros sistemas. El término usuario representa alguien o algo que interactúa con el sistema que se desarrolla. Un caso de uso es un fragmento de funcionalidad que proporciona al usuario un resultado importante (Jacobson, et al., 2000). La unión de todos los casos de uso, conforman el modelo de casos de uso, que describe todas las funcionalidades del sistema, guiando de esta manera el diseño, implementación y prueba, es decir guía el desarrollo de software. (Jacobson, et al., 2000)

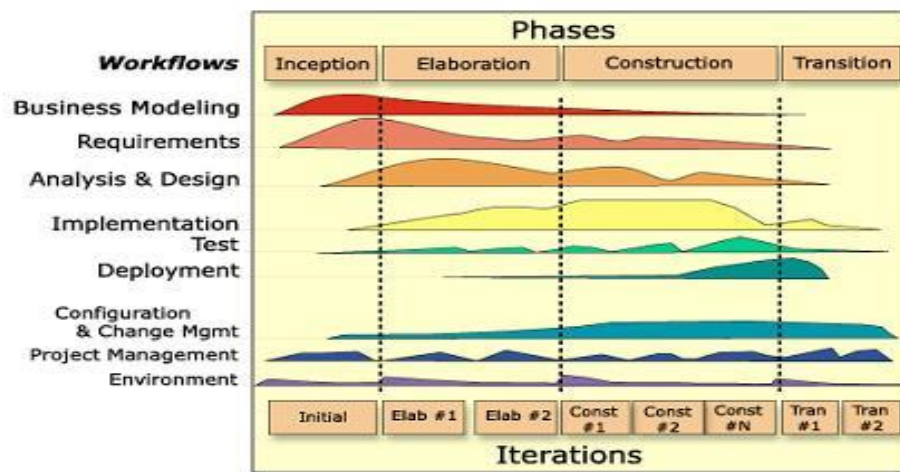


Figure 2 Metodología Proceso Unificado de Rational

Esta metodología es una de las metodologías estándar más eficaz para la implementación y documentación del sistema, pero al ser una metodología tradicional, plantea la creación de una cantidad de documentación excesiva, mucho más de lo que necesita la UNJC y el equipo de desarrollo para tener el Portal correctamente documentado, por lo que RUP no es la metodología que mejor se acopla al proceso de desarrollo del Portal.

1.3.3 Proceso Unificado Ágil

El Proceso Unificado Ágil (en lo adelante **AUP**, por sus siglas en inglés **Agile Unified Process**), es una versión simplificada de RUP, fácil de entender y aplicar, debido a que utiliza varios conceptos vigentes en RUP, adaptados al contexto de una metodología ágil. En AUP, se puede apreciar que los flujos de trabajo con respecto a RUP cambian, dado que el flujo de Modelo, engloba lo que en RUP es Modelado del Negocio, Requerimientos y Análisis y Diseño.

Un producto desarrollado bajo la metodología AUP sigue los siguientes principios básicos:

- El personal sabe lo que se está haciendo: Los integrantes del equipo de desarrollo no necesitan leer documentación de procesos detalladamente, aunque se pueden definir capacitaciones.
- Simplicidad: Todo el proceso de desarrollo se documenta con pocos documentos, de pocas páginas cada uno de ellos.
- Enfocado a las actividades de alto nivel: El proceso se centra en las actividades que realmente cuentan, no en cada detalle posible que podría pasar en un proyecto.

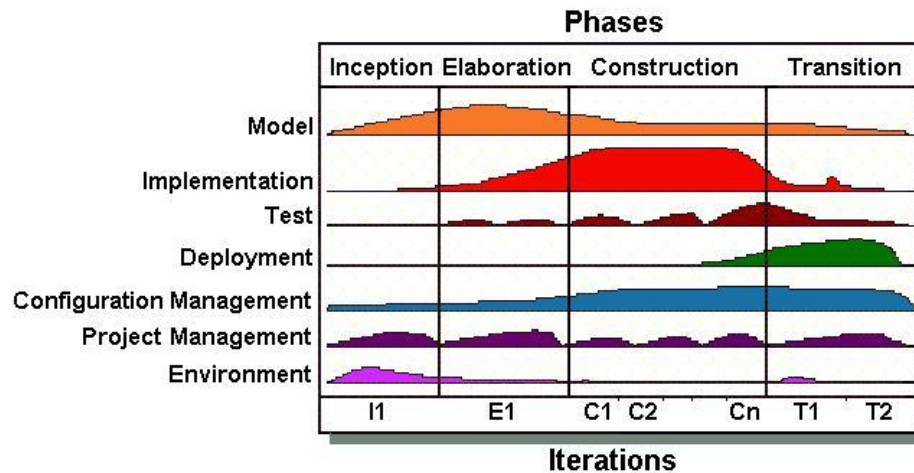


Figure 3 Metodología Proceso Unificado Ágil

Debido a que el Portal debe ser correctamente documentado y que el proceso de desarrollo de software con el uso de una metodología ágil es mejor que con el uso de una metodología tradicional, por las características del equipo de desarrollo, se decide el uso de AUP, como metodología de desarrollo.

1.3.4 Guía Web

El Ministerio Secretaría General de Gobierno de Chile, en el año 2004, crea la Guía para el Desarrollo de Sitios Web de Gobierno Electrónico o Guía Web 1.0, que no es más que una metodología para la creación de sitios web de gobierno electrónico, bajo estándares de contenido y calidad, que recoge la experiencia acumulada tanto en el sector público como el privado, sistematiza y difunde el conocimiento sobre el uso eficiente y eficaz de las nuevas tecnologías de comunicación e información, a través de la promoción y aplicación de un conjunto de recomendaciones y estándares de carácter internacional. (Ministerio Secretaría General del Gobierno de Chile, Enero 2004)

Guía Web 1.0, se centra en brindar un detallado proceso de cómo poner en marcha un sitio web. Está dividida en cinco capítulos los cuales abordan los siguientes temas:

1. Planificación Inicial de un Sitio Web: Explica cómo planificar y organizar el proyecto de desarrollo de un Sitio Web.
2. Definición del Sitio Web: Explica cómo definir los contenidos que se incluirán en un Sitio Web y los pasos para desarrollar la estructura de un sitio y su forma.

3. Diseño Web y Estándares: Explica la importancia de realizar un buen diseño de las páginas del Sitio Web, de acuerdo a normas y estándares.
4. Puesta en Marcha del Sitio Web: Explica los pasos que se deben dar al terminar el desarrollo de un Sitio Web y efectuar su presentación a los usuarios.
5. Plan de Mantenimiento del Sitio Web: Explica los temas referidos a la mantención de un Sitio Web después de su lanzamiento.

En Julio del 2008, se publica la Guía Web 2.0, la cual no es más que una complementación de la primera versión, formando en su conjunto una poderosa metodología de creación de aplicaciones web. Esta versión, se centra en el mantenimiento y optimización de un sitio web en funcionamiento y también está dividida en cinco capítulos, los cuales plantean:

1. Normas que rigen a los Sitios Web: Se refiere a las normas oficiales emitidas por el Gobierno de Chile que deben ser cumplidas y se establecen los contenidos mínimos que deben contener los Sitios Web.
2. Aplicación de Estándares: Se refiere a la definición y aplicación de los estándares sobre construcción de sitios, accesibilidad, interoperabilidad y buenas prácticas que debe cumplir el Sitio Web.
3. Diseño de Interfaces e Interacción: Aborda los elementos necesarios para que las páginas de los Sitios Web cumplan con reglas y estándares de transmisión de contenidos, logrando que la comunicación fluya más directamente entre el Sitio Web y el usuario final.
4. Cómo se llega al Sitio Web: Introduce el concepto de la “Encontrabilidad” que se entiende como la facilidad para que los contenidos de un Sitio Web puedan ser indexados y luego encontrados por sistemas de búsqueda externos e internos.
5. De la Usabilidad a la Utilidad: Presenta el concepto de la Usabilidad como uno de los elementos centrales que deben poseer los Sitios Web, donde se asegure que lo que se publica pueda ser visto y usado sin problemas de acceso por los usuarios.

Como Guía Web es una metodología definida de desarrollo de aplicaciones web y puede ser fácilmente adaptable a las peculiaridades de Cuba u otro país del mundo, se define la utilización de esta metodología en complementación con AUP.

1.4 Lenguajes de Programación

Un lenguaje de programación es utilizado para controlar el comportamiento físico y lógico de una máquina. Permite especificar de una manera más precisa sobre qué datos debe operar una computadora, sobre cómo estos datos deben ser almacenados o transmitidos y qué acciones debe tomar bajo una determinada circunstancia. Está formado por un conjunto de símbolos y reglas sintácticas y semánticas que definen su estructura y el significado de sus elementos y expresiones. (Bonanata, Octubre, 2006)

Debido a las políticas de migración al software libre de la Universidad de las Ciencias Informáticas (en lo adelante UCI) y a las opciones de hosting que posee la UNJC, esta investigación se centra en el estudio de PHP para la implementación en la parte del servidor, XHTML para la construcción de las páginas web, JavaScript como lenguaje en la parte del cliente y de UML para el diseño del sistema.

1.4.1 PHP

PHP es un lenguaje de programación interpretado, diseñado originalmente para la creación de páginas web dinámicas. Es un lenguaje de propósito general ampliamente usado y que está diseñado especialmente para desarrollo web y puede ser embebido dentro de código HTML. Se ejecuta en un servidor web, tomando el código en PHP como su entrada y creando páginas web como salida. Puede ser desplegado en la mayoría de los servidores web y en casi todos los sistemas operativos y plataformas sin costo alguno. (Comunidad Maestros del Web, 2003)

PHP es un acrónimo recursivo que significa **PHP Hypertext Pre-processor**, inicialmente **Personal Home Page Tools**. Fue creado originalmente por Rasmus Lerdorf en 1994 y publicado bajo la PHP License, licencia que es considerada como software libre por la Free Software Foundation. Las principales ventajas que se pueden destacar de PHP son:

Ventajas:

- Es un lenguaje multiplataforma de alto rendimiento.
- Completamente orientado al desarrollo de aplicaciones web dinámicas con acceso a información almacenada en una Base de Datos.

- El código fuente escrito en PHP es invisible al navegador y al cliente ya que es el servidor el que se encarga de ejecutar el código y enviar su resultado HTML al navegador. Esto hace que la programación en PHP sea segura y confiable.
- Capacidad de conexión con la mayoría de los motores de base de datos que se utilizan en la actualidad, destaca su conectividad con MySQL y PostgreSQL.
- Capacidad de expandir su potencial utilizando la enorme cantidad de módulos (llamados ext's o extensiones).
- Posee una amplia documentación en su página oficial, entre la cual se destaca que todas las funciones del sistema están explicadas y ejemplificadas en un único archivo de ayuda.
- Es libre, por lo que se presenta como una alternativa de fácil acceso para todos.
- Permite aplicar técnicas de programación orientada a objetos.
- Biblioteca nativa de funciones sumamente amplia e incluida.
- No requiere definición de tipos de variables aunque sus variables se pueden evaluar también por el tipo que estén manejando en tiempo de ejecución.
- Tiene manejo de excepciones (desde PHP5).

Se hace elección del lenguaje PHP, en su versión estable 5.2.5 o superior, para la implementación del Portal Web.

1.4.2 XHTML

Para entender lo que es el Lenguaje Extensible de Marcado de Hipertexto (en lo adelante **XHTML**, por sus siglas en inglés **eXtensible HyperText Markup Language**) primero se debe definir qué es Lenguaje de Marcado de Hipertexto (en lo adelante **HTML**, por sus siglas en inglés **HyperText Markup Language**) y Lenguaje de Marcas eXtensible (en lo adelante **XML**, por sus siglas en inglés **eXtensible Markup Language**).

HTML es el lenguaje con el que se "escriben" la mayoría de las páginas web. Es un estándar reconocido en todo el mundo y cuyas normas define la World Wide Web Consortium (W3C) para visualizar de forma muy similar una página web en cualquier navegador de cualquier sistema operativo.

XML es un lenguaje de etiquetas desarrollado por el World Wide Web Consortium (W3C). Constituye una simplificación y adaptación para definir la gramática de lenguajes específicos, es decir, una manera de definir lenguajes para diferentes necesidades.

XHTML es una versión más estricta y limpia que se crea con el objetivo de remplazar a HTML ante su limitación de uso con las cada vez más abundantes herramientas basadas en XML. XHTML es una adaptación de HTML al lenguaje XML, por lo que mantiene casi todas sus etiquetas y características, pero añade algunas restricciones y elementos propios de XML. (Eguíluz Pérez, Octubre 2008)

La primera versión se denomina XHTML 1.0 y se publicó el 26 de Enero de 2000. Está orientado al uso de un etiquetado correcto, exige una serie de requisitos básicos a cumplir en lo que a código se refiere. Entre estos requisitos básicos se puede mencionar una estructuración coherente dentro del documento donde se incluirían elementos correctamente anidados, etiquetas en minúsculas, elementos cerrados correctamente, atributos de valores entrecorillados, etc.

1.4.3 JavaScript

JavaScript es un lenguaje de programación que se utiliza principalmente para crear páginas web dinámicas. Una página web dinámica es aquella que incorpora efectos y/o acciones que se activan al pulsar botones u otros elementos con mensajes de aviso al usuario. Es un lenguaje de programación interpretado, por lo que no es necesario compilar los programas para ejecutarlos. En otras palabras, los programas escritos con JavaScript se pueden probar directamente en cualquier navegador sin necesidad de procesos intermedios. A pesar de su nombre, no guarda ninguna relación directa con el lenguaje de programación Java. Legalmente, JavaScript fue creado por Brendan Eich y es una marca registrada de la empresa Sun Microsystems. (Eguíluz Pérez, Mayo 2009)

El código Javascript puede ser integrado dentro de nuestras páginas web. Para evitar incompatibilidades el World Wide Web Consortium (W3C) diseñó un estándar denominado Modelo de Objetos del Documento (**DOM**, por las siglas en inglés **D**ocument **O**bject **M**odel).

Existen al menos tres formas para incluir código JavaScript en las páginas web:

- Incluir el código en la misma página web: El código JavaScript se encierra entre etiquetas `<script>` y se incluye en cualquier zona del documento. Aunque es correcto incluir el bloque de código dentro de la cabecera del documento.
- Definir el código en un archivo externo: Las instrucciones JavaScript se incluyen en un archivo de extensión `*.js` que luego es enlazado a través de etiquetas dentro del código de la página. Es muy útil cuando se trata de reutilizar código para las demás páginas web dentro de un mismo proyecto.
- Incluir el código dentro de elementos de la página web: Consiste en incluir trozos de código JavaScript dentro del código XHTML de la página. Se utiliza, en ocasiones, para definir eventos; puesto que complica el mantenimiento del código.

JavaScript es un lenguaje con muchas posibilidades, permite la programación de pequeños scripts, pero también de programas más grandes, con funciones y estructuras de datos complejas. Además, pone a disposición del programador todos los elementos que forman la página web, para que éste pueda acceder a ellos y modificarlos dinámicamente.

1.4.4 UML

Lenguaje Unificado de Modelado (en lo adelante **UML**, por sus siglas en inglés, **Unified Modeling Language**) es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad. Es un lenguaje de modelado visual que se usa para especificar, visualizar, construir y documentar artefactos de un sistema de software. Captura decisiones y conocimiento sobre los sistemas que se deben construir. Se usa para entender, diseñar, hojear, configurar, mantener, y controlar la información sobre tales sistemas. Está pensado para usarse con todos los métodos de desarrollo, etapas del ciclo de vida, dominios de aplicación y medios. UML incluye conceptos semánticos, notación, y principios generales. Tiene partes estáticas, dinámicas, de entorno y organizativas. Está pensado para ser utilizado en herramientas interactivas de modelado visual que tengan generadores de código así como generadores de informes. (Jacobson, et al., 1999)

Es importante resaltar que UML es un "lenguaje" para especificar y no para describir métodos o procesos. Se utiliza para definir un sistema de software, para detallar los artefactos en el sistema y para documentar y construir. En otras palabras, es el lenguaje en el que está descrito el modelo. Se puede aplicar en una gran variedad de formas para dar soporte a una metodología de desarrollo de software, pero no especifica en sí mismo qué metodología o proceso usar.

Debido a las características antes expuestas hacen propicia la elección de UML en su versión 2.0 como el lenguaje de modelado para la realización del diseño del Portal Web que se va a implementar.

1.5 Sistemas Gestores de Contenidos

Un Sistema Gestor de Contenidos (en lo adelante **CMS**, por sus siglas en inglés **C**ontent **M**anagement **S**ystem) es un software que permite crear una estructura de soporte o framework para la creación y administración de contenidos por parte de los participantes principalmente en páginas web. (García Cuerda, 2004)

Consiste en una interfaz que controla una o varias bases de datos donde se aloja el contenido del sitio. El sistema permite manejar de manera independiente el contenido y el diseño. De esta forma, es posible el trabajo con el contenido y darle si se prefiere, un diseño distinto al original, sin tener que darle formato al contenido de nuevo. Además de permitir la fácil y controlada publicación en el sitio a varios editores.

En la actualidad la gran mayoría de los CMS ofrecen herramientas para generar contenido de forma dinámica, contienen editores de texto los cuales facilitan la actualización, mantenimiento y publicación del contenido, incluyen una manera de insertar los metadatos para una página en específico o para el sitio en general, ofrecen soporte para la optimización del posicionamiento en motores de búsqueda (URL amigables, robots, entre otras) y ofrecen además la obtención de parámetros de usabilidad tales como: la cantidad de visitas al sitio y a un contenido en específico, la búsqueda de contenido dentro del sitio, enlaces a contenidos relacionados, entre otras cualidades.

Muchos CMS también permiten la interacción y colaboración entre los usuarios con el uso de herramientas como foros, blogs, chats y comentarios a contenido, etc., que pueden estar en la instalación por defecto o pueden ser incorporadas posteriormente, debido a que tienen una gran comunidad de desarrolladores que constantemente trabajan en el mantenimiento y la extensión de sus funcionalidades.

El Portal que se va a construir requiere de tecnología multiplataforma y de código abierto para garantizar su utilidad posterior. Por tal motivo la investigación se centra en el análisis de aquellos que brindan soporte para el desarrollo de portales, que sean de código abierto y programados en el lenguaje anteriormente seleccionado PHP 5. Se analizarán a continuación los CMS: Drupal y Joomla!

1.5.1 Drupal

Drupal es un CMS modular multipropósito y muy configurable que permite publicar artículos, imágenes, u otros archivos y servicios añadidos como foros, encuestas, votaciones, blogs y administración de usuarios y permisos. Es un sistema dinámico: en lugar de almacenar sus contenidos en archivos estáticos en el sistema de ficheros del servidor de forma fija, el contenido textual de las páginas y otras configuraciones son almacenados en una base de datos y se editan utilizando un entorno Web. (Comunidad de Usuarios de Drupal, 2005)

Es un programa libre, con licencia GNU/GPL, escrito en PHP, desarrollado y mantenido por una activa comunidad de usuarios. Destaca por la calidad de su código y de las páginas generadas, el respeto de los estándares de la web, y un énfasis especial en la usabilidad y consistencia de todo el sistema.

El diseño de Drupal es especialmente idóneo para construir y gestionar comunidades en Internet. No obstante, su flexibilidad y adaptabilidad, así como la gran cantidad de módulos adicionales disponibles, hace que sea adecuado para realizar muchos tipos diferentes de sitios web.

Drupal se compone de una infraestructura base y un conjunto de módulos que ofrecen un amplio conjunto de funciones, incluyendo sistemas de comercio electrónico, galerías de fotos, administración de listas de correo electrónico, e integración de sistemas de control de versiones. Es posible añadir módulos de terceros para modificar el comportamiento de Drupal y ofrecer nuevas funciones.

El sitio principal de desarrollo y coordinación de Drupal es drupal.org, en el que participan activamente varios miles de usuarios de todo el mundo.

Drupal es un CMS realmente potente en cuanto a rendimiento, flexibilidad, potencia y escalabilidad; las herramientas de construcción de contenido son muy buenas y ofrece una variedad de funcionalidades altamente configurables para la edición del contenido. Sin dudas es muy usado en la actualidad.

1.5.2 Joomla!

Joomla! es un CMS que permite crear sitios web de alta interactividad, profesionalidad y eficiencia. Es una aplicación de código abierto programada en PHP bajo una licencia GPL. La administración de Joomla! está enteramente basada en la gestión en línea de contenidos. Es gestión en línea porque todas las acciones que realizan los administradores de sitios Joomla!, ya sea para modificar, agregar o eliminar contenidos se realiza exclusivamente mediante un navegador web. (Joomla! Spanish, 2007)

Con Joomla! podemos diseñar 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. La interfaz administrativa de es tan sencilla y amigable, que cualquier persona puede administrar sus propios contenidos web sin la necesidad de poseer conocimientos técnicos sobre el software.

Una de las mayores potencialidades que tiene este CMS es la gran cantidad de extensiones existentes programadas por su comunidad de usuarios que aumentan las posibilidades de Joomla con nuevas características y que se integran fácilmente en él. Existen cientos de extensiones disponibles y con diversas funcionalidades.

Después de haber descrito las principales características de Drupal y de Joomla!, existe un difícil dilema a la hora de escoger cuál de los dos sistemas gestores de contenidos es más apropiado para el desarrollo del Portal de la UNJC. Ambas plataformas tienen mucho en común. Son CMS de código abierto y para ambas plataformas, hay cientos de desarrolladores independientes y centros de desarrollo en todo el mundo para atender el desarrollo cada vez mayor en la Web. Ambos, Joomla y Drupal están siendo ampliamente utilizadas por las empresas debido a su naturaleza económica.

Pero a diferencia de Drupal, Joomla está diseñado para los sitios web donde las personas no necesitan gran conocimiento de la tecnología, es más fácil de configurar y poner en marcha con una curva de aprendizaje relativamente baja, es significativamente potente y flexible además de poseer un rendimiento y escalabilidad aceptable para aplicaciones no muy grandes; las funcionalidades de edición y construcción del contenido son excelentes y la consola de administración es mucho más fácil para organizar y encontrar contenido. Joomla! posee además una interfaz amigable en cuanto a diseño y aspecto estético

con una variedad de temas y plantillas muy solidas y profesionales que mejoran el ambiente durante la navegación, tiene un extenso directorio de plugins y componentes y su condición de ser orientado a objetos facilita el desarrollo. Por todas las características anteriormente expuestas, el CMS Joomla!, en su versión 1.6, satisface las necesidades para la construcción del Portal de la UNJC.

1.6 Sistemas Gestores de Base de Datos

Un Sistema Gestor de Bases de Datos (**SGBD**) o Data Base Management System (**DBMS**, por sus siglas en inglés) es una colección de programas cuyo objetivo es servir de interfaz entre la base de datos, el usuario y las aplicaciones. Se compone de un lenguaje de definición de datos, de un lenguaje de manipulación de datos y de un lenguaje de consulta. Un SGBD permite definir los datos a distintos niveles de abstracción y manipular dichos datos, garantizando la seguridad e integridad de los mismos. (Alvarez, 2007)

Un SGBD debe permitir:

- Definir una base de datos: especificar tipos, estructuras y restricciones de datos.
- Construir la base de datos: guardar los datos en algún medio controlado por el mismo SGBD.
- Manipular la base de datos: realizar consultas, actualizarla, generar informes.

Debido a las políticas de migración al software libre de la UCI y a las opciones de hosting que posee la UNJC, esta investigación se centra en el estudio de MySQL como SGBD.

1.6.1 MySQL

MySQL es un SGBD relacional, multihilo, multiplataforma y multiusuario. Es una idea originaria de la empresa Open Source MySQL AB fundada en 1995, que pasó a manos de Sun Microsystems en 2008 cuando adquirió la empresa, luego en 2010 Sun Microsystems fue adquirida por la empresa Oracle Corporation, lo que justifica el desarrollo de MySQL como software libre en un esquema de licenciamiento dual. Por un lado se ofrece bajo la GNU GPL para cualquier uso compatible con esta licencia, pero para aquellas empresas que quieran incorporarlo en productos privativos deben comprar a la empresa una licencia específica que les permita este uso.

EL objetivo que persigue MySQL es cumplir el estándar SQL, pero sin sacrificar velocidad, fiabilidad o usabilidad.

En las últimas versiones se pueden destacar las siguientes características principales:

- El propósito general es lograr mayor velocidad y robustez.
- Soporta gran cantidad de tipos de datos para las columnas.
- Gran portabilidad entre sistemas, puede trabajar en distintas plataformas y sistemas operativos.
- Cada base de datos cuenta con 3 archivos: Uno de estructura, uno de datos y uno de índice y soporta hasta 32 índices por tabla.
- Aprovecha la potencia de sistemas multiproceso, gracias a su implementación multihilo.
- Flexible sistema de contraseñas y gestión de usuarios, con un muy buen nivel de seguridad en los datos.
- El servidor soporta mensajes de error en distintas lenguas.

Ventajas:

- Escalabilidad y flexibilidad: Posibilidad de manipular bases de datos desde un megabyte hasta almacenes de datos enormes y está diseñado para soporte multiplataforma.
- Alto rendimiento: Posee una arquitectura única de motores de bases de datos que permite a los profesionales configurar el servidor MySQL para aplicaciones específicas, dando como resultado un rendimiento espectacular.
- Alta disponibilidad: Solidez y disponibilidad constante ofreciendo una variedad de soluciones desde replicación a servidores de clúster especializados, u ofertas de terceros.
- Robusto soporte transaccional: Posee un soporte completo de ACID (atómica, consistente, aislada, duradera), bloqueo a nivel de filas, posibilidad de transacciones distribuidas, y soporte de transacciones con múltiples versiones.
- Fortaleza en la Web: Posee un motor de consultas de alto rendimiento, con posibilidad de insertar datos a gran velocidad y un buen soporte para funciones web especializadas como las búsquedas texto completo.
- Fuerte protección de datos: Ofrece características de seguridad que aseguran una protección absoluta de los datos.
- Desarrollo de aplicaciones completo: Soporte para procedimientos almacenados, triggers, funciones, vistas, cursores, SQL estándar y otros.
- Facilidades de gestión: Características de gestión automáticas, completa colección de herramientas gráficas de gestión que permiten gestionar, controlar y resolver problemas desde una misma estación de trabajo.

- Código abierto y soporte 24 / 7: Posee un soporte que ofrece una combinación única entre la libertad del código abierto y la confianza de un software con soporte las 24 horas del día y los siete días a la semana.

Se decide el uso de MySQL, en su versión 5.0 como SGBD para el desarrollo del Portal Web.

1.7 Entorno de Desarrollo Integrado para Aplicaciones Web

Un Entorno de Desarrollo Integrado (en lo adelante **IDE**, por sus siglas en inglés **I**ntegrated **D**evelopment **E**nvironment) es un entorno de programación que ha sido empaquetado como un programa de aplicación. Consiste en un editor de código, un compilador, un depurador y en ocasiones un constructor de interfaz gráfica de usuario.

Los IDE's pueden ser aplicaciones por sí solas o pueden ser parte de aplicaciones existentes. Es un editor de código que además puede servir para depurar y facilitar las diferentes tareas necesarias en el desarrollo de cualquier tipo de aplicación.

Como el lenguaje que se va a utilizar para el desarrollo del Portal Web es PHP en su versión estable 5.2.5 o superior, se analizan los IDE's Aptana Studio y Zend Studio.

1.7.1 Aptana Studio

Aptana Studio es un entorno de desarrollo integrado gratuito basado en eclipse y desarrollado por Aptana, Inc., su uso es básicamente para la elaboración de aplicaciones web dinámicas que empleen PHP, es multiplataforma y de código abierto, es una distribución focalizada en el desarrollo web, con soporte a HTML, CSS y Javascript. Aptana Studio contiene también información de soporte para los principales navegadores web: Internet Explorer, Firefox, Opera, Netscape y Safari.

Otras características interesantes de Aptana son las siguientes: (Alvarez, 2007)

- Ayudas visuales para la escritura de scripts en diversos lenguajes, como coloreado y auto escritura del código, ayudas contextuales de referencia a medida que se escribe, etc.
- Visualización de errores de sintaxis a medida que se escribe.
- Debug en Firefox (Debug Internet Explorer también con la versión Profesional).
- Librerías de funciones en Javascript populares en Ajax/Javascript para utilizar en los proyectos.

- Ejemplos ya creados para empezar a conocer las posibilidades de desarrollo rápidamente.
- Explorador de código en forma de árbol.
- Asistente de código para HTML y Javascript.
- Extensión de funcionalidad mediante macros y acciones.
- Servidor local para probar el código.
- Herramientas para trabajo con base de datos.

Todo esto hace de Aptana un programa muy interesante para el desarrollo en la web.

1.7.2 Zend Studio

Zend Studio es el soporte en desarrollos y pruebas de PHP con el set más completo de herramientas para la creación de aplicaciones altamente fiables como lo requiera una empresa. Asegura el desarrollo de software mediante la combinación del IDE con un entorno de prueba que agiliza la seguridad de la calidad, integración y las etapas de los procesos. Brinda todo lo que necesita para construir, probar y entregar aplicaciones de alto rendimiento. (Zend Technologies Ltd., 2007)

Junto con su contraparte Zend Platform, son la propuesta de Zend Technologies para el desarrollo de aplicaciones Web utilizando PHP, actuando Zend Studio como la parte cliente y Zend Platform como la parte servidora. Se trata en ambos casos de software comercial, lo cual contrasta con el hecho de que PHP es software libre. Constituyen el entorno de desarrollo más poderoso para aplicaciones usando PHP, que incluyen características tales como:

- No requiere la instalación previa de PHP y ofrece soporte para PHP 4 y PHP 5.2.x y PHP 5.3.x.
- Resaltado de sintaxis, autocompletado de código, ayuda de código y lista de parámetros de funciones y métodos de clase.
- Ofrece soporte mediante PHPDocs y PHPDocumentor integrado.
- Detección de errores de sintaxis en tiempo real.
- Depuración avanzada: condiciones límites, visualización de errores, vistas avanzadas, variables y buffer de salida.
- Soporte para gestión de grandes proyectos de desarrollo.

- Manual de PHP integrado.
- Soporte para control de versiones.
- Cliente FTP integrado.
- Conectividad e Integración a Bases de Datos: Proporciona la visualización, edición y la capacidad de ejecución para las bases de datos más populares.
- Zend Studio fue diseñado para usarse con el lenguaje PHP; sin embargo ofrece soporte para otros lenguajes Web, como HTML, Javascript y XML.

Se selecciona como IDE para el desarrollo del Portal de la UNJC al Zend Studio en su versión estable 7.1, dado sus características antes expuestas, además de ser concebido con el fin de crear aplicaciones altamente fiables, proporciona una facilidad de uso inigualable, escalabilidad, fiabilidad, y la extensión que los programadores profesionales y de empresas requieren para desarrollar, distribuir, depurar y administrar aplicaciones PHP.

1.8 Herramientas de Ingeniería de Software Asistida por Computadora

Las Herramientas de Ingeniería de Software Asistida por Computadora (en lo adelante **CASE**, por sus siglas en inglés **Computer Aided Software Engineering**) 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. Estas herramientas nos pueden ayudar en todos los aspectos del ciclo de vida de desarrollo del software en tareas como el proceso de realizar un diseño del proyecto, cálculo de costes, implementación de parte del código automáticamente con el diseño dado, compilación automática, documentación o detección de errores entre otras.

1.8.1 Rational Rose

Es una herramienta CASE que cubre todo el ciclo de ayuda de un proyecto: concepción y formalización del modelo, construcción de los componentes y certificación de las distintas fases. Permite trazabilidad real entre modelo (análisis y diseño) y el código ejecutable. (González Blanco, et al., 2003)

Sus principales características son:

- Permite Especificar, Analizar, Diseñar el sistema antes de Codificarlo.

- Mantiene la consistencia de los modelos del sistema software.
- Diseño centrado en casos de uso y enfocado al negocio que genera un software de mayor calidad.
- Chequeo de la sintaxis UML.
- Genera documentación automáticamente.
- Generación de código a partir de los modelos.
- Disponibilidad en múltiples plataformas.
- Capacidades de Ingeniería Inversa.

Rational Rose es una herramienta que se puede encuadrar dentro del grupo de herramientas más técnicas debido a que se encarga de llevar a cabo la automatización de los sistemas para la posterior generación de código. Entre las diferentes ventajas que ofrece el Rational Rose, la más importante es que utiliza la notación estándar en la arquitectura de software UML, permitiendo a los desarrolladores y arquitectos de software utilizar un lenguaje común para poder visualizar el sistema completo. También pueden los diseñadores modelar sus componentes e interfaces en forma individual y luego unirlos con otros componentes del proyecto.

A pesar de que es una herramienta muy recomendada y además profesional, presenta dos grandes inconvenientes, primero es una herramienta propietaria de IBM, empresa la cual radica en los Estados Unidos y a causa del bloqueo hacia Cuba es imposible la compra de una licencia para su uso; segundo, obliga al usuario a desarrollar en máquinas con el sistema operativo propietario como Windows o Macintosh. Por tales motivos no se utilizará esta herramienta en el desarrollo del portal.

1.8.2 Visual Paradigm

Visual Paradigm para UML es una herramienta UML profesional que soporta el ciclo de vida completo del desarrollo de software: análisis y diseño orientados a objetos, construcción, pruebas y despliegue. El software de modelado UML ayuda a una más rápida construcción de aplicaciones de calidad, mejores y a un menor coste. Permite dibujar todos los tipos de diagramas de clases, código inverso, generar código desde diagramas y generar documentación. La herramienta UML CASE también proporciona abundantes tutoriales de UML, demostraciones interactivas de UML y proyectos UML. (Visual Paradigm International Ltd., 2007)

Brinda la posibilidad de generar código a partir de los diagramas, para plataformas como .Net, Java y PHP, así como obtener diagramas a partir de código. Esta es precisamente una gran ventaja puesto que el sistema será desarrollado en PHP. El análisis textual es una técnica útil y práctica para la captura de los requisitos del sistema y la identificación de las clases candidatas, Visual Paradigm es una de las pocas herramientas CASE que soporta el análisis textual.

Tiene disponibilidad en múltiples plataformas y en múltiples versiones. Esta característica es muy importante pues Visual Paradigm está disponible para varios sistemas operativos como Windows, Linux, Unix.

Por tales motivos, la herramienta a utilizar para el modelado de la aplicación es el Visual Paradigm para UML versión 6.4 Enterprise Edition; Suite Update 3.4, debido fundamentalmente a que es una herramienta multiplataforma que ayuda a una rápida construcción de aplicaciones de mejor calidad y a un menor costo y que la UCI posee una licencia para su uso. Permite dibujar todos los tipos de diagramas de clases, soporta la realización de ingeniería tanto directa como inversa, permite control de versiones, presenta una interfaz de uso intuitiva y con muchas facilidades a la hora de modelar los diagramas, además de su robustez, usabilidad y portabilidad.

Conclusiones

Después de analizar las características de la Web 2.0 y haber realizado un estudio preliminar del estado del arte de las principales Metodologías de Desarrollo de Software y de Aplicaciones Web, de los Lenguajes de Programación Web, los Entornos de Desarrollo Integrado, Sistemas Gestores de Contenidos, Sistemas Gestores de Bases de Datos y Herramientas de Ingeniería de Software Asistida por Computadora; teniendo en cuenta las particularidades de la red cubana y las opciones de hospedaje web que posee la UNJC, quedaron definidos para el desarrollo del Portal Web de la UNJC: el uso de la metodología de desarrollo de software AUP y Guía Web en complementación a esta, el lenguaje de programación PHP en su versión estable 5.2.5 o superior, el lenguaje del lado del cliente JavaScript y XHTML v1.0; UML v2.0 como lenguaje de modelado, Zend Studio en su versión 7.1 como entorno de desarrollo, Visual Paradigm para UML 6.4 Enterprise Edition como herramienta CASE, MySQL v5.0 como gestor de bases de datos y Joomla! en su versión estable 1.6 como sistema gestor de contenidos.

CAPÍTULO 2: Características del Portal Web

Introducción

En este capítulo se definen las principales características que debe cumplir el sistema a desarrollar en términos de requisitos funcionales y no funcionales. Se enumeran los requisitos funcionales y no funcionales del sistema, mediante Diagramas de Casos de Uso se describen las relaciones existentes de los actores que se involucran con el sistema y mediante la descripción textual de casos de uso se explica la propuesta de funcionamiento del sistema.

2.1 Descripción del Portal Web.

Una vez realizado un estudio del arte, se puede analizar qué debe hacer el sistema para que se cumplan los objetivos planteados. Para ello se enumeran las prestaciones que el Portal Web será capaz de brindar.

La UNJC, con el uso del Portal podrá:

- Gestionar la publicación de noticias, artículos, informaciones y otras.
- Gestionar los usuarios y permisos de éstos en el Portal.
- Gestionar los eventos de la UNJC.
- Gestionar los servicios que se brindan en la UNJC.
- Gestionar sus estructuras directivas.
- Gestionar sus sociedades científicas.
- Gestionar sus publicaciones.

Estas son las prestaciones centrales del Portal, las cuales habrá que implementar algunas debido a que son específicas para la UNJC. Las prestaciones que se desarrollen, se le añadirán a todas las prestaciones que brinda el CMS Joomla!

2.2 Levantamiento de Requisitos.

Una comunicación efectiva entre los clientes y el equipo de proyecto, con el objetivo de llegar a un entendimiento de lo que hay que hacer, es la clave del éxito en la producción de un software. Los requerimientos del software son las características y cualidades que el sistema debe tener. Estos se dividen en dos grupos, los funcionales y los no funcionales. Una vez definidos e identificados los mismos se tiene la visión general de lo que se quiere hacer en el sistema.

2.2.1 Requisitos Funcionales

Los requisitos funcionales se definen como las condiciones o capacidades que el sistema debe cumplir. Responden a ¿Qué debe hacer el sistema?

Con la realización de entrevistas y reuniones con la dirección de la UNJC y con presidentes de Sociedades Científicas de la UNJC, se identificaron los requerimientos funcionales que debe cumplir el Portal, los cuales se listan a continuación, separados cada uno por el subsistema al que pertenecen.

2.2.1.1 Subsistema Sociedades Científicas:

Este subsistema se encarga de crear, modificar y eliminar las diferentes sociedades científicas de la UNJC, crear modificar y eliminar los blogs pertenecientes a cada **Sociedad Científica** (en lo adelante **SC**), además de encargarse de la gestión del contenido dentro de cada SC.

- RF1 Crear Sociedad Científica
- RF2 Modificar Sociedad Científica
- RF3 Eliminar Sociedad Científica
- RF4 Buscar Sociedad Científica
- RF5 Listar Sociedades Científicas
- RF6 Crear artículo para una SC
- RF7 Modificar artículo de una SC
- RF8 Eliminar artículo de una SC
- RF9 Publicar artículo de una SC
- RF10 Crear Blog de una SC
- RF11 Modificar Blog de una SC
- RF12 Eliminar Blog de una SC
- RF13 Crear categoría de artículos dentro de una SC
- RF14 Modificar categoría de artículos dentro de una SC
- RF15 Eliminar categoría de artículos dentro de una SC
- RF16 Buscar categoría de artículos dentro de una SC
- RF17 Listar todas las categorías de artículos dentro de una SC

2.2.1.2 Subsistema Juntas Directivas:

Este subsistema se encarga de crear, modificar y eliminar las diferentes juntas directivas de la UNJC, la **Junta Directiva Nacional** (en lo adelante **JDN**), las **Juntas Directivas Provinciales** (en lo adelante **JDP**), una para cada provincia y la Isla de la Juventud, además de encargarse de la gestión del contenido dentro de cada **Junta Directiva** (en lo adelante **JD**).

- RF18 Crear Junta Directiva

- RF19 Modificar Junta Directiva
- RF20 Eliminar Junta Directiva
- RF21 Buscar Junta Directiva
- RF22 Crear artículo de una JD
- RF23 Modificar artículo de una JD
- RF24 Eliminar artículo de una JD
- RF25 Publicar artículo de una JD
- RF26 Crear categoría de artículos dentro de una JD
- RF27 Modificar categoría de artículos dentro de una JD
- RF28 Eliminar categoría de artículos dentro de una JD
- RF29 Buscar categoría de artículos dentro de una JD
- RF30 Listar todas las categorías de artículos dentro de una JD

2.2.1.3 Subsistema Administración de Usuarios de la UNJC:

Este subsistema se encarga de crear, modificar y eliminar los datos de los usuarios de la UNJC.

- RF31 Registrar datos del usuario como miembro de la UNJC.
- RF32 Crear Usuario
- RF33 Modificar Usuario
- RF34 Eliminar Usuario
- RF35 Listar Usuarios

2.2.1.4 Subsistema Eventos de la UNJC:

Este subsistema se encarga de crear, modificar y eliminar los diferentes eventos de la UNJC, gestionar las diferentes inscripciones además de encargarse del proceso de suscripción de un usuario a un evento determinado.

- RF36 Solicitar Inscripción a un Evento
- RF37 Crear Evento
- RF38 Modificar Evento
- RF39 Eliminar Evento
- RF40 Buscar Evento
- RF41 Listar Eventos
- RF42 Crear Inscripción a un Evento
- RF43 Modificar Inscripción a un Evento
- RF44 Eliminar Inscripción a un Evento
- RF45 Buscar Inscripción a un Evento
- RF46 Listar Inscripciones a un Evento

2.2.1.5 Subsistema Servicios de la UNJC:

Este subsistema se encarga de gestionar la venta de libros de la UNJC así como el servicio de reservación de comidas, además de encargarse del proceso de petición de uno o varios libros y la reservación de las comidas.

- RF47 Solicitar Libro a la UNJC
- RF48 Reservar comidas
- RF49 Crear Menú de comedor
- RF50 Modificar Menú de comedor
- RF51 Eliminar Menú de comedor
- RF52 Publicar Menú de comedor
- RF53 Crear Información de un Libro
- RF54 Modificar Información de un Libro
- RF55 Eliminar Información de un Libro
- RF56 Buscar Información de un Libro
- RF57 Listar Información de los Libros
- RF58 Crear Reservación de Comedor
- RF59 Modificar Reservación de Comedor
- RF60 Eliminar Reservación de Comedor
- RF61 Buscar Reservación de Comedor
- RF62 Listar Reservaciones de Comedor

2.2.2 Requisitos no Funcionales

Los requisitos no funcionales se definen como las cualidades o propiedades que el software debe tener. Debe pensarse en estas propiedades como las características que hacen al producto atractivo, usable, funcional, rápido o confiable. Normalmente están vinculados a los requerimientos funcionales, es decir, una vez que se conozca lo que el sistema debe hacer, podemos determinar cómo ha de comportarse. Se definen entonces los siguientes requerimientos no funcionales.

2.2.2.1 Usabilidad**Tipo de Usuario Final:**

- RnF1 El sistema puede ser usado por cualquier persona sin discapacidad física, incluso aquellas con pocos conocimientos informáticos.
- RnF2 El sistema será fácil de manejar por cualquier persona, incluso aquellas con pocos conocimientos informáticos.

Tipo de Aplicación Informática

- RnF3 El Sistema a desarrollar es un Sitio Web bajo las características de la Web 2.0, que responda a las necesidades de la UNJC.

Finalidad

- RnF4 Desarrollar un Portal Web que pueda ofrecer mayor facilidad para compartir información, mayor interoperabilidad y colaboración en línea, además de permitir a sus usuarios una visualización activa de la información que se publica.

Ambiente

- RnF5 El sistema debe estar optimizado para los navegadores Mozilla Firefox 3 o superior, Internet Explorer 7 o superior y Opera 9 o superior.
- RnF6 El sistema debe funcionar correctamente en todos los tamaños de ventanas de los navegadores Web.
- RnF7 Las etiquetas de campos deben presentarse de manera normalizada en todas las páginas, deben aparecer con la misma tipografía y el mismo puntaje.
- RnF8 El sistema debe explicar la entrada de datos, con ejemplos concisos y sintéticos, en los campos que sean complejos.
- RnF9 El sistema debe presentar en todas las páginas un Mapa del Sitio y este debe tener vínculos a las diferentes secciones que muestra.
- RnF10 El sistema debe mostrar las rutas de acceso según la navegación que tenga el usuario (Migas de Pan).
- RnF11 El sistema debe tener acceso al menú principal desde cualquiera de sus páginas.
- RnF12 Se debe ubicar en cada página el Pie de Página.
- RnF13 El Pie de Página debe mostrar elementos como los derechos de autor, nombre de la institución, preguntas y respuestas más frecuentes (FAQs), términos y condiciones de uso, etc.
- RnF14 El sistema debe tener enlaces a otros sitios afines relacionados con la entidad.
- RnF15 El sistema debe mostrar de una forma distintiva mensajes para cada eventualidad que ocurra con la entrada al sistema.
- RnF16 El sistema debe mostrar textos personalizados para los mensajes de error.
- RnF17 Las funcionalidades del sistema deberán estar agrupadas dentro de la interfaz según los datos sobre los que actúan.

2.2.2.2 Confiabilidad

- RnF18 El Sistema debe estar disponible para su utilización cada vez que el usuario lo requiera. Solamente debe cesar la disponibilidad cuando sea necesario realizar mantenimiento en el mismo o los servidores donde se encuentre instalado no estén disponibles.
- RnF19 En caso de fallo de energía en la PC cliente, el Sistema, al reanudarse las operaciones, debe enviar o actualizar la información al/del servidor para continuar el trabajo de forma estable.
- RnF20 En caso de fallo de energía en la PC servidor, el Sistema, al reanudarse las operaciones, debe enviar o actualizar la información al cliente para continuar el trabajo de forma estable.
- RnF21 En caso de interrupción del servicio de red en la PC cliente, el Sistema, al reanudarse las operaciones, debe enviar o actualizar la información al/del servidor para continuar el trabajo de forma estable.
- RnF22 En caso de interrupción del servicio de red en la PC servidor, el Sistema, al reanudarse las operaciones, debe enviar o actualizar la información al cliente para continuar el trabajo de forma estable.
- RnF23 En caso de interrupción de los servicios, el Sistema debe informar a sus usuarios y debe resolver el problema lo antes posible, en un margen de 24 horas.
- RnF24 Al ocurrir una excepción el Sistema debe informar al usuario y volver a un estado anterior de funcionamiento estable.

2.2.2.3 Eficiencia

- RnF25 El tiempo de carga del Sistema debe ser de un máximo permisible de 5 segundos.
- RnF26 Los tiempos de respuestas por transacción deben ser como máximo de 20 segundos al igual que la velocidad de procesamiento de la información.
- RnF27 El Sistema debe guardar en caché las páginas de contenido ya cargadas para agilizar la navegación en el Portal.
- RnF28 El Sistema debe utilizar la menor cantidad de páginas para ejecutar todas las funciones posibles, es decir, agrupar funciones afines en las mismas páginas.
- RnF29 Las Imágenes deben tener un tamaño menor a 1 megabyte (MB) y en formatos tales como: *.png, *.gif, *.jpg o *.jpeg.
- RnF30 El consumo de memoria y recursos será bajo, en el orden de pocos megabytes.

2.2.2.4 Soporte

- RnF31 Instalación, reparación o actualización de módulos en caso de que el Sistema lo requiera o sea necesario.
- RnF32 Se debe realizar la transferencia tecnológica de la aplicación a través de la entrega de documentación digital y recursos presenciales.
- RnF33 Se debe facilitar un manual de usuario para consultas de las personas que administrarán el Sistema.
- RnF34 Servicio de Hospedaje Web (web-hosting) facilitado por el MIC.

2.2.2.5 Restricciones de Diseño

- RnF35 El lenguaje de programación a utilizar será PHP 5.2.x.
- RnF36 El CMS a utilizar es Joomla! 1.6.x.
- RnF37 El SGBD a utilizar es MySQL 5.0.

2.2.2.6 Requisitos para la documentación de usuarios en línea y ayuda del sistema.

- RnF38 Ayuda para la configuración del sistema. Visualización de breves descripciones asociadas a los módulos más relevantes y vínculos a las configuraciones más representativas como permisos y configuración básica.

2.2.2.7 Requisitos de Licencia

- RnF39 Se hará uso de la licencia GNU/GPL (Licencia Pública General de GNU).

2.3 Casos de Uso del Sistema

Como ya se había planteado anteriormente, un caso de uso es un fragmento de funcionalidad que le brinda al usuario una información importante para él, llámese usuario una persona u otro sistema externo a la aplicación. Cuando se unen todos los casos de uso, se conforma el diagrama de casos de uso, que contiene toda la funcionalidad del sistema. Es necesario tener en cuenta que para cada requisito funcional se debe identificar un caso de uso, aunque se pueden agrupar los casos de uso con funcionalidad común o separar los casos de uso complejos, y también se pueden aplicar patrones de casos de uso para el mejor entendimiento del diagrama. Un patrón muy usado en todo sistema es el de crear, obtener, actualizar y eliminar algún elemento, al cual se le llama **CRUD** (por sus siglas en inglés **C**reate, **R**etrieve, **U**pdate, **D**estroy).

Update and Delete). En este trabajo, para mejorar el entendimiento del diagrama, en vez de situar “CRUD Elemento”, se decidió el uso de la palabra gestionar, para darle la forma “Gestionar Elemento”. Los casos de uso identificados en el Portal son:

- 1) Crear Artículo de una SC.
- 2) Modificar Artículo de una SC.
- 3) Eliminar Artículo de una SC.
- 4) Buscar Artículo de una SC.
- 5) Publicar Artículo de una SC.
- 6) Gestionar Categoría de una SC.
- 7) Gestionar Sociedades Científicas.
- 8) Gestionar Blog de cada Sociedad Científica.
- 9) Crear Artículo de una JD.
- 10) Modificar Artículo de una JD.
- 11) Eliminar Artículo de una JD.
- 12) Buscar Artículo de una JD.
- 13) Publicar Artículo de una JD.
- 14) Gestionar Categoría de una JD.
- 15) Gestionar JD.
- 16) Registrar Datos como usuario de la UNJC.
- 17) Gestionar Usuario.
- 18) Solicitar Inscripción en Evento.
- 19) Gestionar Eventos.
- 20) Gestionar Inscripciones a Eventos.
- 21) Solicitar Libros.
- 22) Reservar comidas.
- 23) Gestionar Menú de comedor.
- 24) Gestionar Información de libros.
- 25) Gestionar Reservaciones de Comedor.

2.3.1 Diagrama de Casos de Uso del Sistema

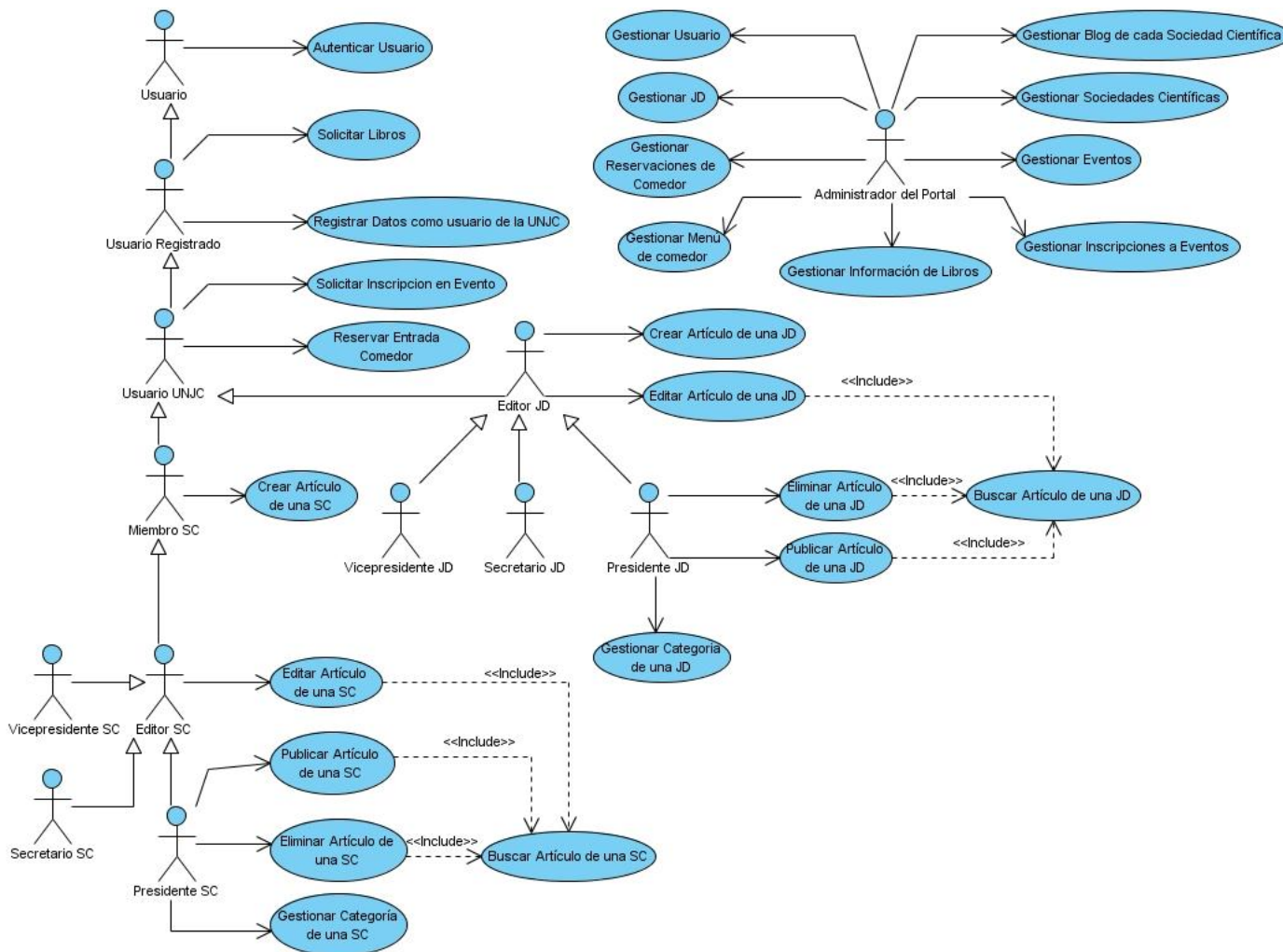


Figure 4 Diagrama de Casos de Uso del Sistema

2.3.2 Actores del Sistema

Los actores del sistema como anteriormente se explicó, son personas o sistemas externos que interactúan con el Portal. Aquí se describirán cada uno de los actores del sistema y su objetivo en el portal.

Actor	Descripción
Administrador del Portal	Es la persona con todos los permisos en el sistema para gestionar las Sociedades Científicas, las Juntas Directivas, Usuarios, Permisos, etc.

Miembro SC	Crea artículos para publicar en una SC.
Editor SC	Persona que crea y/o edita los artículos creados por miembros de una SC.
Secretario SC	Crea y/o edita los artículos creados por los miembros de una SC.
Vicepresidente SC	Crea y/o edita los artículos creados por los miembros de una SC.
Presidente SC	Crea, modifica, elimina y publica los artículos de una SC y gestiona las categorías dentro de la SC.
Editor JD	Crea y/o edita los artículos de una Junta Directiva.
Secretario JD	Crea y/o edita los artículos de una Junta Directiva.
Vicepresidente JD	Crea y/o edita los artículos de una Junta Directiva.
Presidente JD	Crea, modifica, elimina y publica los artículos de una JD y gestiona las categorías dentro de la JD.
Usuario	Autenticarse en el sistema.
Usuario UNJC	Son los usuarios que pertenecen a la UNJC.
Usuario Registrado	Otros Usuarios que se registran en el sistema.

2.3.3 Descripción de los Casos de Uso del Sistema

La descripción textual de los casos de uso es lo que explica el funcionamiento del sistema completo. Aquí se hará un resumen de las descripciones textuales de los casos de uso identificados en el Portal.

2.3.3.1 Caso de Uso Gestionar Sociedades Científicas.

Objetivo	Crear, modificar y eliminar las Sociedades Científicas de la UNJC.
Actores	Administrador del Portal: (Inicia) Crea, modifica, elimina, obtiene y lista las Sociedades Científicas de la UNJC.
Resumen	Este caso de uso inicia cuando el Administrador del Portal decide crear, modificar, eliminar, ver o listar una Sociedad Científica de la UNJC.
Complejidad	Alta

Prioridad	Crítico	
Precondiciones	El Administrador del Portal está registrado en el sistema y ha sido autenticado como Administrador del Portal.	
Postcondiciones	Queda creada, modificada o eliminada una Sociedad Científica.	
Flujo de eventos		
Flujo básico Gestionar Sociedades Científicas		
	Actor	Sistema
1.	Decide gestionar las Sociedades Científicas.	
2.		Lista todas las Sociedades Científicas existentes en la UNJC.
3.	Decide tomar una acción sobre las Sociedades Científicas.	<ul style="list-style-type: none"> - Crear Sociedad Científica. Ver sección 1: “Crear Sociedad Científica”. - Modificar Sociedad Científica. Ver sección 2: “Modificar Sociedad Científica”. - Eliminar Sociedad Científica. Ver sección 3: “Eliminar Sociedad Científica”. - Buscar Sociedad Científica. Ver sección 4: “Buscar Sociedad Científica”.
Sección 1: “Crear Sociedad Científica”		
Flujo básico Crear Sociedad Científica.		
	Actor	Sistema
1.	Selecciona crear una nueva Sociedad Científica.	
2.		Muestra un formulario con los datos necesarios para crear la Sociedad Científica.
3.	Entra los datos para crear la Sociedad	

	Científica y los envía.	
4.		Valida que los datos entrados sean correctos.
5.		Valida que los datos entrados no estén repetidos.
6.		Guarda la Sociedad Científica.
7.		Envía mensaje de confirmación de que todo ocurrió correctamente.
8.		Termina el caso de uso.

Flujos alternos

Nº 5 Los datos entrados están repetidos

	Actor	Sistema
1.		Muestra un mensaje avisando que la información entrada está repetida.
2.		Comienza por el Nº 2 del Flujo normal de eventos.

Nº 6 No se guardó la Sociedad Científica.

	Actor	Sistema
1.		Muestra un mensaje avisando de que ha habido un error al guardar la Sociedad Científica.
2.		Comienza por el Nº 2 del Flujo normal de eventos.

Sección 2: “Modificar Sociedad Científica”

Flujo básico Modificar Sociedad Científica

	Actor	Sistema
1.	Selecciona la Sociedad Científica a	

	modificar.	
2.		Verifica que haya seleccionado al menos una Sociedad Científica.
3.		Carga los datos de la Sociedad Científica seleccionada y las muestra en un formulario.
4.	Modifica los datos necesarios y los envía.	
5.		Valida que los datos entrados sean correctos.
6.		Valida que los datos entrados no estén repetidos.
7.		Guarda la Sociedad Científica.
8.		Envía mensaje de confirmación de que todo ocurrió correctamente.
9.		Termina el caso de uso.

Flujos alternos

Nº 2 No se seleccionó ninguna Sociedad Científica

	Actor	Sistema
1.		Muestra un mensaje avisando de que debe seleccionar una Sociedad Científica.
2.		Comienza por el Nº 1 del Flujo normal de eventos.

Nº 6 Los datos entrados están repetidos

	Actor	Sistema
1.		Muestra un mensaje avisando que la información entrada está repetida.
2.		Comienza por el Nº 3 del Flujo

		normal de eventos.
Nº 7 No se guardó la Sociedad Científica.		
	Actor	Sistema
1.		Muestra un mensaje avisando de que ha habido un error al guardar la Sociedad Científica.
2.		Comienza por el Nº 3 del Flujo normal de eventos.
Sección 3: “Eliminar Sociedad Científica”		
Flujo básico Eliminar Sociedad Científica		
	Actor	Sistema
1.	Selecciona las Sociedades Científicas a eliminar y envía la solicitud para eliminarlas.	
2.		Verifica que haya seleccionado al menos una Sociedad Científica.
3.		Elimina la o las Sociedades Científicas seleccionadas.
4.		Envía mensaje de confirmación de que todo ocurrió correctamente.
5.		Termina el caso de uso.
Flujos alternos		
Nº 2 No seleccionó ninguna Sociedad Científica		
	Actor	Sistema
1.		Muestra un mensaje avisando de que debe seleccionar al menos una Sociedad Científica.
2.		Comienza por el Nº 1 del Flujo normal de eventos.

Nº 3 No se eliminó la Sociedad Científica.		
	Actor	Sistema
1.		Muestra un mensaje avisando de que ha habido un error al eliminar la o las Sociedades Científicas seleccionadas.
2.		Comienza por el Nº 1 del Flujo normal de eventos.
Sección 4: “Buscar Sociedad Científica”		
Flujo básico Buscar Sociedad Científica		
	Actor	Sistema
1.	Entra los datos por los que quiere buscar una Sociedad Científica determinada.	
2.		Busca la Sociedad Científica según los datos entrados.
3.		Muestra la o las Sociedades Científicas que cumplen con los parámetros entrados.
4.		Termina el caso de uso.
Relaciones	CU Incluidos	
	CU Extendidos	
Requisitos no funcionales		
Asuntos pendientes		

Conclusiones

En este capítulo se listaron los requerimientos funcionales y no funcionales del sistema y se describieron algunos de los casos de uso identificados, por lo que la descripción del sistema está completa y se puede comenzar la implementación del mismo, debido a que se conoce qué debe hacer el sistema.

CAPÍTULO 3: Implementación y Prueba del Portal Web

Introducción

En este capítulo se describe el funcionamiento del CMS Joomla! en su versión 1.6, se hace referencia a los patrones arquitectónicos que se utilizan en Joomla! y en las extensiones desarrolladas, se implementan las funcionalidades que requiere el Portal y se describen las pruebas funcionales y no funcionales a las extensiones construidas y se dan a conocer los resultados de dichas pruebas.

3.1 El CMS Joomla!

Para explicar la arquitectura del CMS Joomla!, es preciso introducir dos patrones arquitectónicos que son usados por Joomla! y además son unos de los más usados en el mundo del software; estos son el Patrón N-Capas y el Patrón Modelo-Vista-Controlador (en lo adelante **MVC**, por sus siglas en ingles **Model-View-Controller**).

3.1.1 Patrón N-Capas

El Patrón de arquitectura en N-Capas ayuda a estructurar las aplicaciones que puedan ser descompuestas en grupos de sub tareas, en el cual cada grupo de sub tareas es un nivel particular de abstracción. (Buschmann, et al., 1996)

Este Patrón descompone la aplicación en cuantas capas sean necesarias para distribuir las sub tareas, donde el cliente interactúa con la capa más alta y ésta a su vez interactúa con su capa inmediata inferior, así hasta llegar a la capa más baja, como se puede observar en la siguiente imagen.

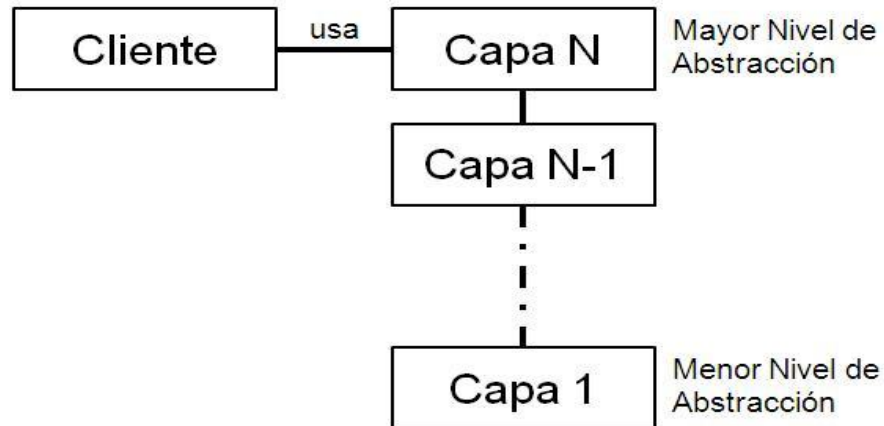


Figure 5 Patrón de Arquitectura N-Capas

3.1.2 Patrón Modelo-Vista-Controlador

Actualmente las aplicaciones de software permiten un alto nivel de interacción con el usuario, donde la meta principal es mantener el negocio de la aplicación completamente independiente de la interfaz de usuario, por lo que el Patrón MVC divide una aplicación interactiva en tres componentes:

- El Modelo que contiene la funcionalidad principal y los datos.
- La Vista que muestra la información al usuario.
- El Controlador que maneja la interacción con el usuario y la entrada de datos.

La Vista y el Controlador componen la interfaz de usuario. (Buschmann, et al., 1996)

Cuando el usuario realiza una acción, ésta es recibida por el Controlador, el cual decide cuál Modelo utilizar, qué datos necesita del Modelo y también decide qué Vista utilizar para mostrar los resultados.

3.1.3 Arquitectura del CMS Joomla!

La arquitectura de Joomla! es la de un sistema tres capas, donde:

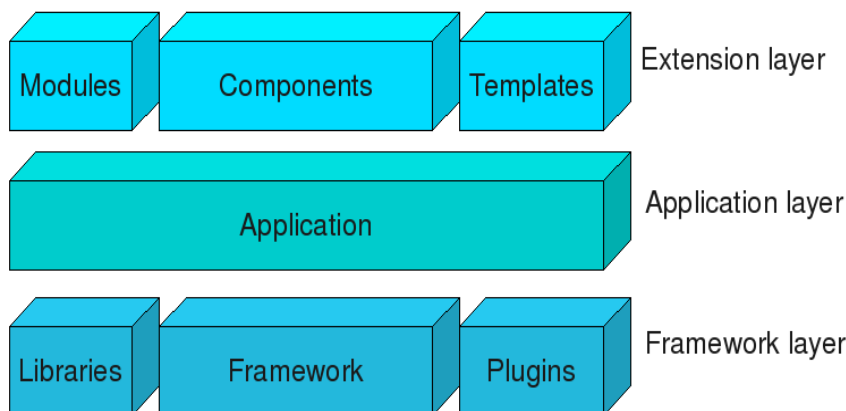


Figure 6 Arquitectura de Joomla!

- La capa inferior, llamada “**Framework Layer**”, consiste en el Framework propio de Joomla!, las Librerías que son utilizadas por este Framework o que son instaladas para su uso por otros desarrolladores y los Plugins que extienden la funcionalidad de dicho Framework.
- La capa del medio, llamada “**Application Layer**”, consiste en las aplicaciones que extienden la clase *JApplication* del Framework, las cuales son:
 - *JInstallation* que se encarga del proceso de instalación del CMS.
 - *JAdministrator* que es la encargada de la parte de administración del sitio.
 - *JSite* que es la encargada de la parte pública del sitio.
- La capa superior, llamada “**Extension Layer**”, consiste en las extensiones al Framework de Joomla! y sus aplicaciones, cuyas extensiones son Módulos, Componentes y Plantillas que se instalan en el núcleo del sistema.

Los Módulos son bloques pequeños que disponen información o herramientas, son como mini-utilidades que pueden ubicarse en cualquier lugar del sitio y se asignan a una posición definida en la plantilla. A veces funcionan en conjunción con algún componente y otras veces son independientes.

Los Componentes son bloques más o menos grandes que crean un menú propio en el panel de administración y casi siempre requiere de una configuración previa. Un componente puede llevar a cabo cualquier operación no contemplada por las funciones generales del código del núcleo, como manejar datos, mostrar configuraciones, proveer funciones, entre otras. Los componentes trabajan mano a mano

con módulos y plug-ins lo que hace posible que se pueda expandir ampliamente la funcionalidad de Joomla!

Las Plantillas son las encargadas de definir el aspecto estético, la disposición de los elementos y el estilo visual del sitio web.

También son consideradas extensiones los paquetes de idioma que posibilitan mostrar las interfaces de usuario en el idioma local preferido. Estos paquetes no tienen ningún impacto sobre el contenido actual del sitio como son los artículos, comentarios, etc.

3.1.4 Estructura de un Componente del CMS Joomla!

Una de las principales características de Joomla!, es que los componentes son desarrollados utilizando el Patrón MVC. Un componente es dividido en dos partes, una parte para mostrar el contenido que es ejecutada en el sitio, la cual se guarda en el directorio */components* y la otra parte es para la administración del contenido, del lado de la administración, la cual se guarda en el directorio */administrator/components*. Cada una de estas partes, es una implementación del Patrón MVC, por lo que básicamente, un componente contiene dos subsistemas, uno para la administración y el otro para el sitio.

3.2 Implementación

La implementación es la parte del proceso de desarrollo de software que se ocupa de la creación del sistema en términos de componentes, es decir ficheros de código fuente, scripts, ejecutables y similares. (Jacobson, et al., 2000)

Para satisfacer las necesidades de la UNJC, fue necesaria la implementación de dos componentes, la modificación de un tercero y la creación de una plantilla.

3.2.1 Componente Juntas Directivas

Este componente se encarga de crear, modificar y eliminar las diferentes juntas directivas de la UNJC, la JDN, las JDP, una para cada provincia y la Isla de la Juventud, además de encargarse de la gestión del contenido dentro de cada JD.

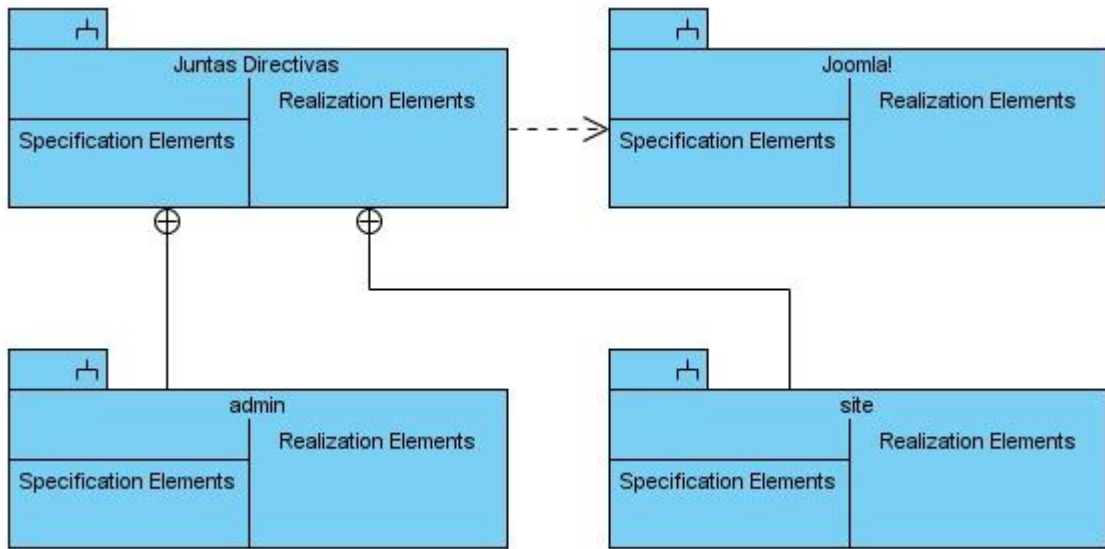


Figure 7 Subsistema Juntas Directivas

Cada uno de los subsistemas contenidos en el componente Juntas Directivas, tiene la siguiente estructura.

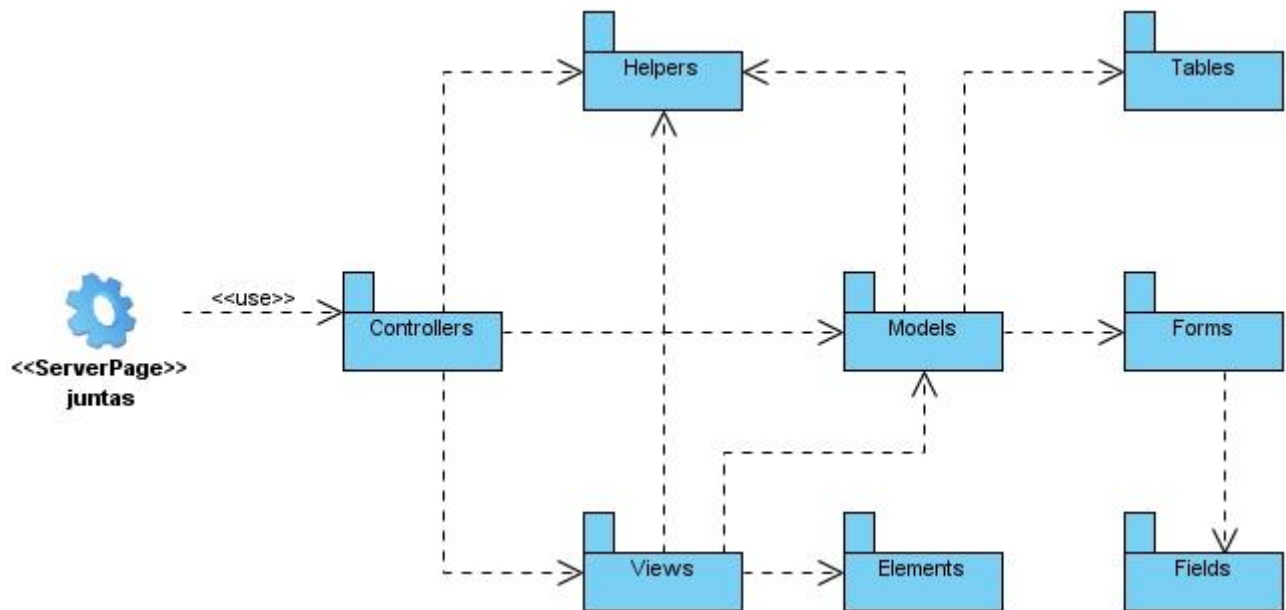


Figure 8 Diseño del Subsistema Juntas Directivas

3.2.2 Componente Sociedades Científicas

Este componente se encarga de crear, modificar y eliminar las diferentes sociedades científicas de la UNJC, crear modificar y eliminar los blogs pertenecientes a cada SC, además de encargarse de la gestión del contenido dentro de cada SC.

El componente Sociedades Científicas tiene la misma estructura que el componente explicado anteriormente.

3.2.3 Componente ChronoForms

Este es un componente desarrollado por terceros, que tuvo que ser modificado para satisfacer las necesidades de la UNJC. Este componente es un constructor de formularios dinámico, el cual crea formularios completamente con solo arrastrar y soltar los elementos que desea que los conformen. Este componente puede restringir el acceso al formulario por medio de grupos de usuarios, por lo que se hizo un cambio para que restringiera el acceso por medio de niveles de acceso.

El otro cambio que se realizó es que los formularios creados por este componente, no verifican si los datos a llenar, tienen una capacidad limitada, por lo que se implementó esa funcionalidad para darle límite a un formulario determinado.

Para hacer efectivo dicho cambio se introdujo código adicional al núcleo del componente, el código se validó y quedó definido como se muestra a continuación.

```
//obtiene la capacidad
$cantidad = $MyForm->form_params->get('cantidad');
if ($cantidad > 0) {
    $db = JFactory::getDbo();

    //obtiene la tabla y el campo para medir la capacidad
    list($tabla, $campo) = explode(".", $MyForm->form_params->get('tabla_cantidad'));
    $query = '';
    if (isset($campo)) {
        $query = 'SELECT SUM('.$tabla.'.'.$campo.') AS cantidad FROM '.$tabla;
    } else {
        $query = 'SELECT COUNT(*) AS cantidad FROM '.$tabla;
    }
    $db->setQuery($query);
    $object = $db->loadObject();
    if($cantidad <= $object->cantidad)
    {
        echo "Ya la reservacion esta llena";
        return ;
    }
}
```

Figure 9 Ejemplo de Código implementado para el Componente Chronoforms

3.3 Pruebas

Con las pruebas se verifica el resultado de la implementación, probando cada componente por separado y luego en su conjunto como un sistema listo a entregar. (Jacobson, et al., 2000)

Existen dos tipos de métodos de prueba los cuales son:

3.3.1 Pruebas de caja blanca

Las pruebas de caja blanca están dirigidas principalmente a las funciones internas del sistema. Se realizan probando los caminos lógicos del sistema y examinando el estado del programa en varios puntos para determinar si el estado real coincide con el esperado. Como las pruebas de caja blanca están hechas para probar cada uno de los caminos que puede tomar el código fuente, en aplicaciones grandes no resulta práctico el uso de estas pruebas, a no ser utilizarlas sólo en las funcionalidades más importantes.

3.3.2 Pruebas de caja negra

Las pruebas de caja negra se desarrollan sobre la interfaz del software. Estas pruebas se encargan de verificar que las funciones que debe desempeñar el sistema son operativas. Se centran en los Requisitos

Funcionales del software, sin interesarse por el funcionamiento interno del mismo. La realización de estas pruebas permite encontrar:

- Funciones incorrectas.
- Errores de interfaz.
- Errores en las salidas.
- Errores en el acceso a los datos.

3.3.3 Pruebas a los componentes desarrollados

Para garantizar que los componentes desarrollados realizan su trabajo correctamente, se les realizaron las pruebas funcionales correspondientes. A continuación se muestra un resumen de los casos de prueba realizados.

3.3.3.1 Caso de Prueba Gestionar Juntas Directivas

Descripción General

Caso de Uso Gestionar Juntas Directivas: Crear, modificar y eliminar las Juntas Directivas de la UNJC.

Condiciones de Ejecución

El Administrador del Portal está registrado en el sistema y ha sido autenticado como Administrador del Portal.

SC 1 Crear Junta Directiva

Escenario	Descripción	Título	Alias	Padre	Estado	Acceso	Lenguaje	Descripción	Respuesta del sistema	Resultado de la Prueba
EC 1.1 Crear Junta Directiva	Se creará una Junta Directiva	N/A	N/A	N/A	N/A	N/A	N/A	N/A	Se crea la Junta Directiva y muestra un mensaje de confirmación	Satisfactoria
		varchar	varchar	Lista desplegable	Lista desplegable	Lista desplegable	Lista desplegable	Text		
EC 1.2 Los datos entrados están repetidos	Se entra el alias de la Junta Directiva de una Junta ya existente	NA	NA	NA	NA	NA	NA	NA	El sistema debe mostrar un mensaje donde diga que el alias entrado ya está en uso.	Satisfactoria
		varchar	varchar	Lista desplegable	Lista desplegable	Lista desplegable	Lista desplegable	Text		
EC 1.3 No se guardó la Junta Directiva.	Cuando ocurre un error en la Base de Datos al crear la Junta Directiva	NA	NA	NA	NA	NA	NA	NA	El sistema debe mostrar un mensaje donde diga que la Junta Directiva no se pudo guardar correctamente	Satisfactoria
		varchar	varchar	Lista desplegable	Lista desplegable	Lista desplegable	Lista desplegable	Text		

Figure 10 SC Crear Junta Directiva

SC 2 Modificar Junta Directiva

Escenario	Descripción	Título	Alias	Padre	Estado	Acceso	Lenguaje	Descripción	Selección	Respuesta del sistema	Resultado de la Prueba
EC 1.1 Modificar Junta Directiva	Se modificará una Junta Directiva	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	Se modifica la Junta Directiva y muestra un mensaje de confirmación	Satisfactoria
		varchar	varchar	Lista desplegable	Lista desplegable	Lista desplegable	Lista desplegable	Text	Selección		
EC 1.2 No se seleccionó ninguna Junta Directiva.	No se seleccionó una Junta Directiva.	NA	NA	NA	NA	NA	NA	NA	N/A	El sistema debe mostrar un mensaje donde diga que se debe seleccionar una Junta Directiva.	Satisfactoria
		varchar	varchar	Lista desplegable	Lista desplegable	Lista desplegable	Lista desplegable	Text	Selección		
EC 1.3 Los datos entrados están repetidos.	Se entra el alias de la Junta Directiva de una Junta ya existente	NA	NA	NA	NA	NA	NA	NA	N/A	El sistema debe mostrar un mensaje donde diga que el alias entrado ya está en uso.	Satisfactoria
		varchar	varchar	Lista desplegable	Lista desplegable	Lista desplegable	Lista desplegable	Text	Selección		
EC 1.3 No se guardó la Junta Directiva.	Cuando ocurre un error en la Base de Datos al modificar la Junta Directiva	NA	NA	NA	NA	NA	NA	NA	N/A	El sistema debe mostrar un mensaje donde diga que la Junta Directiva no se pudo guardar correctamente	Satisfactoria
		varchar	varchar	Lista desplegable	Lista desplegable	Lista desplegable	Lista desplegable	Text	Selección		

Figure 11 SC Modificar Junta Directiva

SC 3 Eliminar Junta Directiva

Escenario	Descripción	Selección	Respuesta del sistema	Resultado de la Prueba
EC 1.1 Eliminar Junta Directiva	Se eliminará una o varias Juntas Directivas	N/A	Se elimina la Junta Directiva o las Juntas Directivas seleccionadas y muestra un mensaje de confirmación	Satisfactoria
		Selección		
EC 1.2 No se seleccionó ninguna Junta Directiva	No se seleccionó una Junta Directiva.	N/A	El sistema debe mostrar un mensaje donde diga que se debe seleccionar una Junta Directiva.	Satisfactoria
		Selección		
EC 1.3 No se eliminó la Junta Directiva.	Cuando ocurre un error en la Base de Datos al eliminar la Junta Directiva.	N/A	El sistema debe mostrar un mensaje donde diga que no se pudo eliminar la Junta Directiva.	Satisfactoria
		Selección		

Figure 12 SC Eliminar Junta Directiva

SC 4 Buscar Junta Directiva

Escenario	Descripción	Filtro	Nivel Máximo	Estado	Acceso	Lenguaje	Respuesta del sistema	Resultado de la Prueba
EC 1.1 Buscar Junta Directiva	Se buscará una o varias Juntas Directivas según el criterio de búsqueda	N/A	N/A	N/A	N/A	N/A	Se buscan y muestran las Juntas Directivas que cumplan con el criterio de búsqueda.	Satisfactoria
		varchar	Lista desplegable	Lista desplegable	Lista desplegable	Lista desplegable		

Figure 13 SC Buscar Junta Directiva

3.3.3.1 Caso de Prueba Gestionar Sociedades Científicas

Descripción general

Caso de Uso Gestionar Sociedades Científicas: Crear, modificar y eliminar las Sociedades Científicas de la UNJC.

Condiciones de ejecución

El Administrador del Portal está registrado en el sistema y ha sido autenticado como Administrador del Portal.

SC 1 Crear Sociedad Científica

Escenario	Descripción	Título	Alias	Padre	Estado	Acceso	Lenguaje	Descripción	Respuesta del sistema	Resultado de la Prueba
EC 1.1 Crear Sociedad Científica	Se creará una Sociedad Científica	N/A	N/A	N/A	N/A	N/A	N/A	N/A	Se crea la Sociedad Científica y muestra un mensaje de confirmación	Satisfactoria
		varchar	varchar	Lista desplegable	Lista desplegable	Lista desplegable	Lista desplegable	Text		
EC 1.2 Los datos entrados están repetidos	Se entra el alias de la Sociedad Científica de una Sociedad ya existente	NA	NA	NA	NA	NA	NA	NA	El sistema debe mostrar un mensaje donde diga que el alias entrado ya está en uso.	Satisfactoria
		varchar	varchar	Lista desplegable	Lista desplegable	Lista desplegable	Lista desplegable	Text		
EC 1.3 No se guardó la Sociedad Científica.	Cuando ocurre un error en la Base de Datos al crear la Sociedad Científica	NA	NA	NA	NA	NA	NA	NA	El sistema debe mostrar un mensaje donde diga que la Sociedad Científica no se pudo guardar correctamente	Satisfactoria
		varchar	varchar	Lista desplegable	Lista desplegable	Lista desplegable	Lista desplegable	Text		

Figure 14 SC Crear Sociedad Científica

SC 2 Modificar Sociedad Científica

Escenario	Descripción	Título	Alias	Padre	Estado	Acceso	Lenguaje	Descripción	Selección	Respuesta del sistema	Resultado de la Prueba
EC 1.1 Modificar Sociedad Científica	Se modificará una Sociedad Científica	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	Se modifica la Sociedad Científica y muestra un mensaje de confirmación	Satisfactoria
		varchar	varchar	Lista desplegable	Lista desplegable	Lista desplegable	Lista desplegable	Text	Selección		
EC 1.2 No se seleccionó ninguna Sociedad Científica	No se seleccionó una Sociedad Científica.	NA	NA	NA	NA	NA	NA	NA	N/A	El sistema debe mostrar un mensaje donde diga que se debe seleccionar una Sociedad Científica.	Satisfactoria
		varchar	varchar	Lista desplegable	Lista desplegable	Lista desplegable	Lista desplegable	Text	Selección		
EC 1.3 Los datos entrados están repetidos	Se entra el alias de la Sociedad Científica de una Sociedad ya existente	NA	NA	NA	NA	NA	NA	NA	N/A	El sistema debe mostrar un mensaje donde diga que el alias entrado ya está en uso.	Satisfactoria
		varchar	varchar	Lista desplegable	Lista desplegable	Lista desplegable	Lista desplegable	Text	Selección		
EC 1.3 No se guardó la Sociedad Científica.	Cuando ocurre un error en la Base de Datos al modificar la Sociedad Científica	NA	NA	NA	NA	NA	NA	NA	N/A	El sistema debe mostrar un mensaje donde diga que la Sociedad Científica no se pudo guardar correctamente	Satisfactoria
		varchar	varchar	Lista desplegable	Lista desplegable	Lista desplegable	Lista desplegable	Text	Selección		

Figure 15 SC Modificar Sociedad Científica

SC 3 Eliminar Sociedad Científica

Escenario	Descripción	Selección	Respuesta del sistema	Resultado de la Prueba
EC 1.1 Eliminar Sociedad Científica	Se eliminará una o varias Sociedades Científicas	N/A	Se elimina la Sociedad Científica o las Sociedades Científicas seleccionadas y muestra un mensaje de confirmación	Satisfactoria
		Selección		
EC 1.2 No se seleccionó ninguna Sociedad Científica	No se seleccionó una Sociedad Científica.	N/A	El sistema debe mostrar un mensaje donde diga que se debe seleccionar una Sociedad Científica.	Satisfactoria
		Selección		
EC 1.3 No se eliminó la Sociedad Científica.	Cuando ocurre un error en la Base de Datos al eliminar la Sociedad Científica	N/A	El sistema debe mostrar un mensaje donde diga que no se pudo eliminar la Sociedad Científica.	Satisfactoria
		Selección		

Figure 16 SC Eliminar Sociedad Científica

SC 4 Buscar Sociedad Científica

Escenario	Descripción	Filtro	Nivel Máximo	Estado	Acceso	Lenguaje	Respuesta del sistema	Resultado de la Prueba
EC 1.1 Buscar Sociedad Científica	Se buscará una o varias Sociedades Científicas según el criterio de búsqueda	N/A	N/A	N/A	N/A	N/A	Se buscan y muestran las Sociedades Científicas que cumplan con el criterio de búsqueda.	Satisfactoria
		varchar	Lista desplegable	Lista desplegable	Lista desplegable	Lista desplegable		

Figure 17 SC Buscar Sociedad Científica

3.3.4 Pruebas de Rendimiento

Para las pruebas de rendimiento, específicamente pruebas de Carga y Estrés, que se le realizó al Portal Web, se usó la herramienta JMeter versión 2.3.4 que no es más que una herramienta Java desarrollada dentro del proyecto Jakarta, que permite realizar pruebas de Rendimiento y Pruebas Funcionales sobre Aplicaciones Web.

JMeter es un software de código abierto, un 100% pura aplicación de escritorio Java diseñado para pruebas de carga de comportamiento funcionales y la medición del rendimiento. Originalmente fue diseñado para probar las aplicaciones Web, pero desde entonces se ha expandido a otras funciones de prueba. Es utilizado para probar el rendimiento tanto en los recursos estáticos como dinámicos. Puede ser utilizado para simular una carga pesada en un servidor, de red o un objeto para poner a prueba su resistencia o para analizar el rendimiento general en diferentes tipos de carga. Puede usarse además para hacer un análisis gráfico de rendimiento o para probar el comportamiento de su servidor / script / objeto con carga pesada concurrentes. (Apache Software Foundation , 2001)

Para la realización de las pruebas se hizo necesario tener en cuenta las condiciones del escenario, tanto del hardware como software, donde se encuentra la aplicación; para obtener una correcta información de comportamiento y resultados en general. Por tanto se hizo necesario simular las pruebas en un escenario con las características siguientes:

Hardware:

- Tipo de procesador: Pentium (R) Dual-Core CPU E5400, 2.70GHz (2CPUs)
- Memoria: 2014 MB RAM
- Tipo de Red: Ethernet 10/100Mbps

Software:

- Tipo de Servidor Web: Apache v2.2.6
 - Memoria máxima: 32 MB por hilo de ejecución.
 - Máximo de hilos concurrentes: 20, 50, 100 y 500 (respectivamente para cada escenario de prueba).
- Plataforma: SO. MicroSoft Windows XP Professional (5.1, Build 2600).
- Servidor de BD: MySQL.

➤ Lenguaje: PHP

En un ambiente con las características anteriormente especificadas se obtuvo que para un total de 20 usuarios (hilos), la aplicación generó un total de 27.4 Kb/seg de transferencia de datos lo que incurrió en un rendimiento de 1.1 seg. Se demuestra que la aplicación es estable ya que para estos 20 usuarios se mantuvo prestando servicios todo el tiempo sin incurrir en fallos.

De forma similar, con las características del escenario descrito, se realizaron pruebas para distintas cantidades de usuarios, en las cuales la aplicación generó un promedio de 31.1 Kb/seg de transferencia de datos lo que provocó un rendimiento de 2.6 seg como promedio. La aplicación se mantuvo prestando servicio durante casi todo el tiempo con un margen de fallo de 1.2 %, que no es significativo lo que demuestra que la aplicación es un 98.8% estable.

Conclusiones

En este capítulo se describieron la estructura y el funcionamiento del CMS Joomla! y se describieron las características de sus componentes, así como los patrones arquitectónicos utilizados. Se implementaron dos componentes y se modificó un tercero para satisfacer las necesidades de la UNJC. También se les realizaron pruebas a los componentes desarrollados y a la aplicación en general para garantizar su correcto funcionamiento, por lo que el sistema entra en su capacidad operacional inicial.

CONCLUSIONES

Una vez concluida la investigación, se ha determinado que los objetivos planteados se cumplieron satisfactoriamente y se arribó a las siguientes conclusiones:

- Se realizó un estado del arte donde se determinó que el uso de la metodología AUP, conjuntamente con Guía Web, son las más adecuadas para el desarrollo y documentación del Portal y el uso del CMS Joomla! en su versión 1.6 el cual es la mejor opción para su construcción.
- Se licitaron y especificaron todos los requerimientos funcionales y no funcionales mediante los cuales se conocieron las principales funcionalidades y cualidades que satisfacen las necesidades de la UNJC.
- Se identificaron las extensiones a implementar para satisfacer las necesidades de la UNJC y mediante su diseño se obtuvo una comprensión más detallada y se dio soporte a todos los requerimientos.
- Se implementaron las extensiones dándole solución a los requerimientos planteados.
- Se diseñaron, implementaron y documentaron las pruebas realizadas a las extensiones desarrolladas y se verificó que funcionan correctamente.
- La documentación generada posibilitó el desarrollo del Portal y ayudará a la creación de futuras ampliaciones u otras modificaciones.

El Portal Web de la UNJC se encuentra listo para su puesta en marcha y explotación, ofreciendo herramientas que permiten la creación, edición y publicación de contenido de forma dinámica, obtención de parámetros de usabilidad y contribuye a una mayor interoperabilidad y colaboración en línea de los usuarios.

RECOMENDACIONES

Los autores recomiendan:

- Mantener una cercana información acerca de las actualizaciones del CMS Joomla! y todas las extensiones instaladas en el Portal.
- Cambiar el componente de carro de compra por el conocido carro de compra VirtueMart, dado que todavía no existe una versión estable compatible con Joomla! 1.6.
- Cambiar el componente de creación de formularios dinámicos por el componente ckForms, dado que todavía no existe una versión estable compatible con Joomla! 1.6.
- Dado que en el portal se maneja toda la información de las Juntas Directivas y las Sociedades Científicas, se recomienda que cada una de ellas tenga su propio portal y que se eliminen esas responsabilidades del Portal Web construido.
- Revisar muy de cerca cualquier cambio que ocurra en el CMD Joomla! con respecto al manejo de las categorías y los artículos y reflejar dichos cambios automáticamente en los componentes de Juntas Directivas y Sociedades Científicas.
- Mejorar la plantilla en general con el uso de AJAX, Flash y otras tecnologías las cuales no fueron utilizadas por limitaciones tecnológicas de Cuba con respecto al ancho de banda de la infraestructura de red.
- Implementar servicios web y servicios de RSS que no se desarrollaron por limitaciones tecnológicas de Cuba con respecto al ancho de banda de la infraestructura de red.

REFERENCIAS BIBLIOGRÁFICAS

¿Qué es Web 2.0? **O'Reilly, Tim. 2004.** San Francisco, California, USA : O'Reilly Media, 2004.

Alvarez, Miguel Angel. 2007. DesarrolloWeb.com. [Online] Noviembre 16, 2007. [Cited: Enero 18, 2011.] [Disponible en <http://www.desarrolloweb.com/articulos/aptana-studio.html>]. <http://www.desarrolloweb.com>.

—. **2001.** DesarrolloWeb.com. [Online] Mayo 9, 2001. [Cited: Enero 8, 2011.] [Disponible en <http://www.desarrolloweb.com/articulos/392.php>]. <http://www.desarrolloweb.com>.

Alvarez, Sara. 2007. DesarrolloWeb.com. [Online] Julio 31, 2007. [Cited: Enero 16, 2011.] [Disponible en <http://www.desarrolloweb.com/articulos/sistemas-gestores-bases-datos.html>]. <http://www.desarrolloweb.com>. N/A.

Angela Byron, Jeff Robbins, Addison Berry, James Walker. Diciembre, 2008. *Using Drupal*. Primera Edición. United States of America : O'Reilly Media, Inc., Diciembre, 2008. 978-0-596-51580-5.

Apache Software Foundation . 2001. The Apache Jakarta Project. [Online] Apache Software Foundation , 2001. [Cited: Junio 1, 2011.] [Disponible en <http://jakarta.apache.org/jmeter/>]. <http://jakarta.apache.org>.

Beck, Kent. 1999. *Extreme Programming Explained*. 1999. [Publicado por primera vez en Septiembre 29, 1999]. 0201616416.

Bonanata, Maximiliano. Octubre, 2006. *Programación y Algoritmos*. Buenos Aires, Argentina : MP Ediciones S.A., Octubre, 2006.

Buschmann, Frank, et al. 1996. *Pattern-Oriented Software Architecture*. West Sussex : John Wiley & Sons Ltd, 1996. Vol. I. 0-471-95889-7.

Byron, Angela and Berry, Addison. 2009. *Drupal*. España : Grupo Anaya Comercial, 2009. 9788441526129.

Campderrich Falgueras, Benet. 2003. *Ingeniería del software*. Barcelona, España : Editorial UOC, 2003. 84-8318-997-6.

Centro Nacional de Información de Ciencias Médicas. 2009. Infomed. Red de Salud de Cuba. [Online] Marzo 31, 2009. [Cited: Enero 12, 2011.] <http://www.infomed.sld.cu/>. N/A.

Comunidad de Usuarios de Drupal. 2005. Drupal Hispano. [Online] Abril 11, 2005. [Cited: Enero 12, 2011.] [Disponible en <http://drupal.org.es/drupal>]. <http://drupal.org.es>.

Comunidad Maestros del Web. 2003. Maestros del Web. [Online] Julio 21, 2003. [Cited: Enero 8, 2011.] [Disponible en <http://www.maestrosdelweb.com/editorial/phpintro/>]. <http://www.maestrosdelweb.com>.

Costa, Joan. 2009. *DirCom, Estratega de la complejidad*. Valencia, España : Universitat de València, 2009. 978-84-370-7202-9.

Cunningham, Ward. 1987. *Explicacion de concepto "Wiki"*. 1987.

Departamento de Lenguajes y Sistemas Informáticos, Universidad de Sevilla. 2004. *Introducción a las Aplicaciones Web*. Sevilla, España : s.n., 2004.

Eguíluz Pérez, Javier. Mayo 2009. *Introducción a JavaScript*. Mayo 2009. [Disponible en <http://www.librosweb.es>].

— **Octubre 2008.** *Introducción a XHTML*. Octubre 2008. [Disponible es <http://www.librosweb.es/xhtml/>].

Freeman, Linton. 2004. *The Development of Social Network Analysis*. Vancouver, BC Canada : Empirical Press, 2004. [Disponible en: <http://aris.ss.uci.edu/~lin/book.pdf>]. 1-59457-714-5.

García Cuerda, Xavier. 2004. *Introducción a los Sistemas de Gestión de Contenidos (CMS) de código abierto. Tecnologías y Comunicación Multimedias*. [Online] Septiembre 29, 2004. [Cited: Enero 8, 2011.] [Disponible en *Introducción a los Sistemas de Gestión de Contenidos (CMS) de código abierto*].

González Blanco, Rubén and Pérez Tobalina, Sergio. 2003. *Introducción a Rational Rose. Funcionalidad General*. Departamento de Lenguajes y Sistemas Informáticos, Universidad Politécnica de Cataluña. Barcelona, España : Facultad de Informatica de Barcelona, 2003. Presentación. N/A.

Hernando, Roberto. 2009. *rhernando.net*. [Online] 2009. [Cited: Junio 2, 2011.] Blog personal de Roberto Hernando. http://www.rhernando.net/modules/tutorials/doc/ing/met_soft.html.

Jacobson, Ivar, Booch, Grady and Rumbaugh, James. 1999. *El Lenguaje Unificado de Modelado. Manual de Referencia*. España : Addison Wesley, 1999. [Disponible en <http://bibliodoc.uci.cu/pdf/reg03050.pdf>].

—. **2000.** *El Proceso Unificado de Desarrollo de Software*. España : Addison Wesley, 2000. 0-201-57169-2.

Joomla! 2007. Joomla! Características generales. *Gold Scripts soluciones inteligentes para webmasters exitosos*. [Online] Noviembre 14, 2007. [Cited: Enero 12, 2011.] [Disponible en <http://www.goldscripsts.com/2007/11/14/joomla-caracteristicas-generales/>]. <http://www.goldscripsts.com>.

Joomla! Spanish. 2007. Manual de Usuario Joomla! Spanish. [Online] 2007. [Cited: Enero 12, 2011.] [Disponible en <http://ayuda.joomlaspanish.org/ayuda-joomla/>]. <http://ayuda.joomlaspanish.org>.

López Ramot, Jordi. 2008. *Mashup, sumando en la red*. Zaragoza, España : Ibercaja Zentrum, 2008. [Disponible en <http://www.ibercajalav.net/img/mashups.pdf>]. N/A.

2009. Manual de PHP. [Online] 2009. [Cited: Enero 8, 2011.] <http://www.manualdephp.com>.

Mendoza Sanchez, María A. 2004. Informatizate. Metodologías de Desarrollo de Software. [Online] Junio 7, 2004. [Cited: Enero 20, 2011.] [Disponible en http://www.informatizate.net/articulos/metodologias_de_desarrollo_de_software_07062004.html]. <http://www.informatizate.net>. N/A.

Ministerio Secretaría General del Gobierno de Chile. Enero 2004. *Guía para Desarrollo de Sitios Web, Guía Web 1.0*. Primera Edición. Santiago de Chile, Chile : ANDROS , Enero 2004. [Disponible en <http://www.guiaweb.gob.cl>]. N/A.

—. **Junio 2007.** *Guía para Desarrollo de Sitios Web, Guía Web 2.0*. Santiago de Chile, Chile : s.n., Junio 2007. [Disponible en <http://www.guiaweb.gob.cl>].

Nafría, Ismael. 2008. *La Web 2.0. El usuario, el nuevo rey de Internet*. Barcelona, España : Gestión 2000, 2008. 978-84-9875-003-4.

Orihuela, José Luis. 2006. *La revolución de los blogs*. Madrid, España : La Esfera de los Libros, 2006. 84-9734-498-7.

The International Comission of Jurists. 1991. International Commission of Jurists, The Rule of Law and Human Rights. [Online] 1991. [Cited: Enero 12, 2011.] <http://www.icj.org>.

The PHP Group. 2010. PHP. *PHP Website Copyright*. [Online] The PHP Group, Diciembre 22, 2010. [Cited: Enero 8, 2011.] <http://php.net>.

REFERENCIAS BIBLIOGRÁFICAS

Portal Web de la Unión Nacional de Juristas de Cuba

Unión Nacional de Juristas de Cuba. 1977. Unión Nacional de Juristas de Cuba. *Portal de la UNJC.* [Online] Junio 8, 1977. [Cited: 1 14, 2011.] <http://www.unjc.co.cu>.

Visual Paradigm International Ltd. 2007. Visual Paradigm for UML. [Online] Marzo 5, 2007. [Cited: Enero 5, 2011.] [Disponible en http://www.freedownloadmanager.org/es/downloads/Paradigma_Visual_para_UML_%5Bcuenta_de_Plataforma_de_Java_14715_p/]. <http://www.visual-paradigm.com>.

Warp Networks S.L. 2008. Warp Sharing Innovation. *Warp Networks S.L.* [Online] 2008. [Cited: 2 16, 2011.] [Disponible en <http://warp.es/mysql/productos/razones>]. <http://warp.es>.

WordPress.org. 2007. WordPress Características generales. *Gold Scripts soluciones inteligentes para webmasters exitosos.* [Online] Noviembre 14, 2007. [Cited: Enero 12, 2011.] [Disponible en <http://www.goldscripsts.com/2007/11/14/wordpress-caracteristicas-generales/>]. <http://www.goldscripsts.com>.

World Wide Web Consortium España. 2010. World Wide Web Consortium España. *Sitio Web de la W3C Oficina España.* [Online] Mayo 6, 2010. [Cited: Enero 12, 2011.] <http://www.w3c.es/divulgacion/guiasbreves>. N/A.

Zend Technologies Ltd. 2007. *Zend Studio.* Ramat Gan, Israel : CONCATEL S.L., 2007.

APÉNDICES

Glosario de Términos

1. **Sitio Web:**- Es una colección de páginas web relacionadas, imágenes, vídeos u otros archivos digitales típicamente comunes a un dominio de Internet o subdominio en Internet. Una página web es un documento HTML/XHTML accesible generalmente mediante el protocolo HTTP de Internet, a través de un URL raíz común llamado portada, que normalmente reside en el mismo servidor físico.
2. **Interoperabilidad:**- Es la condición mediante la cual sistemas heterogéneos pueden intercambiar procesos o datos, la capacidad que tiene un producto o un sistema, cuyas interfaces son totalmente conocidas, para funcionar con otros productos y eso sin restricción de acceso o de implementación.
3. **World Wide Web:**- La World Wide Web o comúnmente llamada Web es un sistema de distribución de información basado en hipertexto o hipermedios enlazados y accesibles a través de Internet. Con un navegador web, un usuario visualiza sitios web compuestos de páginas web que pueden contener texto, imágenes u otros contenidos multimedia, y navega a través de ellas usando hiperenlaces.
4. **Diseño Centrado en el Usuario:**- Es una filosofía de diseño que tiene por objetivo la creación de productos que resuelvan necesidades concretas de sus usuarios finales, consiguiendo la mayor satisfacción y mejor experiencia de uso posible con el mínimo esfuerzo por su parte.
5. **Visualización pasiva:**- Se refiere a cuando un usuario no puede crear ni editar contenido en un sitio web, pues visualiza el contenido de forma “pasiva”, ya que sólo ve el contenido que se le ofrece y no puede realizar ningún cambio.
6. **Visualización activa:**- Se refiere a cuando un usuario puede crear y editar contenido en un sitio web, pues visualiza el contenido de forma “activa”, ya que puede ver el contenido que se le muestra y también puede editarlo o crear nuevo contenido.
7. **Framework:**- (en español Marco de Trabajo) define, en términos generales, un conjunto estandarizado de conceptos, prácticas y criterios para enfocar un tipo de problemática particular, que sirve como referencia para enfrentar y resolver nuevos problemas de índole similar. Es una estructura conceptual

y tecnológica de soporte definida, normalmente con artefactos o módulos de software concretos, con base en la cual otro proyecto de software puede ser organizado y desarrollado.

8. **Base de Datos:**- Conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso. Una base de datos relacional es una base de datos que cumple con el modelo relacional, que no es más que un modelo de datos basado en la lógica de predicados y en la teoría de conjuntos que su idea fundamental es el uso de “relaciones”.
9. **Multiusuario:**- Característica de un programa que permite proveer servicio y procesamiento a múltiples usuarios simultáneamente (tanto en paralelismo real como simulado). Multiusuarios son todos los sistemas que cumplen simultáneamente las necesidades de dos o más usuarios, que comparten los mismos recursos
10. **Multihilo:**- Un hilo de ejecución o subproceso es una característica que permite a una aplicación realizar varias tareas a la vez (concurrentemente). Los distintos hilos de ejecución comparten una serie de recursos tales como el espacio de memoria, los archivos abiertos, situación de autenticación, etc. Un hilo es básicamente una tarea que puede ser ejecutada en paralelo con otra tarea. Los sistemas multihilos son aquellos que sostienen varios hilos de ejecución simultáneamente.
11. **SQL:**- El lenguaje de consulta estructurado (**SQL** por sus siglas en inglés Structured Query Language) es un lenguaje declarativo de acceso a bases de datos relacionales que permite especificar diversos tipos de operaciones en éstas. Una de sus características es el manejo del álgebra y el cálculo relacional permitiendo efectuar consultas con el fin de recuperar -de una forma sencilla- información de interés de una base de datos, así como también hacer cambios sobre ella.
12. **URL:**- (en español, Localizador Uniforme de Recursos) es una secuencia de caracteres, de acuerdo a un formato modélico y estándar, que se usa para nombrar recursos en Internet para su localización o identificación.