



Universidad de las Ciencias Informáticas

Facultad 1

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

***Portal Web de la Unión de Jóvenes Comunistas para
la Universidad de las Ciencias Informáticas***

Autor: Yennifer Delgado Mesa

Tutores: Ing. Cecilia Esther Hernández Espinosa

Ing. Yuneldis Reyes Velázquez

La Habana, Cuba, junio de 2015

“Año 57 de la Revolución”

Declaro ser la autora de la presente tesis y reconocer a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

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

Yennifer Delgado Mesa

Firma del Autor

Ing. Cecilia Esther Hernández Espinosa

Firma del Tutor

Ing. Yuneldis Reyes Velázquez

Firma del Tutor



Lo fundamental es que seamos capaces de hacer cada día algo que perfeccione lo que hicimos el día anterior.

A mi mamita, por ser como eres, buena, sencilla, franca, exigente, enorme de corazón. Por tu infinito amor, por tu apoyo y dedicación, por ser mi mejor amiga, porque sin ti esto no fuera posible.

A mi papá por alegrarme en los momentos tristes y difíciles, por apoyarme y comprenderme. Por haber creído y confiado en mí en todo momento. Gracias por ser el mejor padre del mundo.

A mi hermanita por quererme tanto y estar siempre conmigo, por haber compartido mis llantos y alegrías, por haberme dado siempre mucha confianza y fuerza para seguir adelante. Te quiero mucho.

A mi familia en general que siempre me apoyó. Sin su amor y preocupación no hubiese llegado a ser lo que soy.

A mi novio por haber estado junto a mí en los momentos buenos y malos, por su comprensión y cariño, porque siempre creíste en mí, por tu inagotable paciencia y amor incondicional. Por ser el hombre con el que siempre soñé.

A mis suegros: Arelis y Rolando y mi cuñada Lorena, por hacerme sentir parte de su familia.

A mi amiga Iliana, por todas las cosas que pasamos juntas, por apoyarme, por darme aliento a seguir pese a las dificultades.

A los amigos que conocí aquí en la UCI y que han sido de gran apoyo.

A todas las personas que de una forma u otra hicieron posible este sueño.

A mis abuelos por la experiencia.

A mis padres, por la luz.

A mis amigos -todos- por la confianza.

A mi novio, por ese dulce rescate.

Resumen

El desarrollo que han alcanzado las tecnologías en los últimos años ha marcado un cambio significativo en el manejo de la información en varias ramas de la sociedad; destacándose su inclusión en la divulgación de esta y permitiendo que llegue a los interesados de manera rápida y directa. En este sentido, la promoción de las actividades o eventos de la Unión de Jóvenes Comunistas (UJC) en la Universidad de las Ciencias Informáticas (UCI) presenta deficiencias, puesto que la información de la organización solo es conocida a través de reuniones o por correo electrónico, lo que dificulta la disponibilidad e integridad de la misma. Con el objetivo de solucionar esta situación el presente trabajo propone desarrollar un portal web para la UJC de la UCI.

Para la realización del portal se utilizó PostgreSQL 9.1 como gestor de Base de Datos y Apache 2.2 como servidor web. La metodología de desarrollo de software a seguir será OpenUP. El Visual Paradigm en su versión 8.0 será la herramienta con la que se realizara el modelado de los procesos y artefactos arrojados por dicha metodología y como sistema de gestión de contenidos se utilizará Drupal en su versión 7.34. Para la validación del sistema se realizaron varias pruebas funcionales, pruebas de rendimiento, pruebas de seguridad y usabilidad, las cuales arrojaron un conjunto de no conformidades que fueron resueltas en iteraciones posteriores. El sistema desarrollado será de beneficio para los miembros de la organización ya que permitirá consultar información relevante de forma sencilla, rápida y mediante una interfaz simple y amigable.

Palabras claves: gestión de información, información, portal web, UJC.

Índice

Introducción	1
Capítulo 1. Fundamentación Teórica	5
1.1 Conceptos asociados al dominio del problema.....	5
1.2 Historia y funcionamiento de la UJC.....	5
1.3 Arquitectura de Información.....	6
1.4 Análisis de los sitios web homólogos.....	6
1.4.1 Sistemas internacionales.....	7
1.4.2 Sistemas nacionales	9
1.5 Sistema de Gestión de Contenidos (CMS).....	11
1.6 Metodología de desarrollo.....	13
1.7 Tecnologías y herramientas a utilizar en la implementación.....	16
1.7.1 Lenguajes de programación.....	16
1.7.2 Sistema gestor de Base de Datos	17
1.7.3 Entorno integrado de desarrollo	18
1.7.4 Servidor de aplicaciones web.....	19
1.7.5 Herramienta Case	19
1.7.6 Herramientas de pruebas de software.....	20
1.8 Conclusiones Parciales.....	20
Capítulo 2. Características del sistema	21
2.1 Propuesta del sistema.....	21

2.2	Modelo de Dominio.....	21
2.3	Levantamiento de requisitos.....	22
2.3.1	Requisitos Funcionales	22
2.3.2	Requisitos no Funcionales	27
2.4	Definición de los Casos de Uso del Sistema.....	29
2.4.1	Actores del sistema	29
2.4.2	Casos de uso del sistema	30
2.4.3	Diagrama de Casos de Uso del Sistema.....	31
2.4.4	Descripción de Casos de Uso del Sistema.....	31
2.5	Arquitectura de Drupal.....	33
2.6	Patrones de diseño.....	34
2.7	Diagrama de Clases del Diseño.....	36
2.8	Diagrama de Secuencia.....	38
2.9	Modelo de Despliegue.....	40
2.10	Conclusiones Parciales.....	41
Capítulo 3. Implementación y Prueba del Sistema		42
3.1	Implementación.....	42
3.1.1	Diagrama de componentes	42
3.1.2	Descripción de componentes	43
3.2	Estándares de codificación.....	44
3.3	Pruebas.....	45
3.3.1	Funcionales.....	46

3.3.2	Seguridad.....	49
3.3.3	Carga	51
3.3.4	Estrés.....	52
3.3.5	Usabilidad	52
3.4	Pantallas de la aplicación.....	54
3.5	Conclusiones del capítulo.....	55
	Conclusiones	56
	Recomendaciones	57
	Referencias.....	60
	Bibliografía.....	64
	Glosario de Términos.....	67

Índice de Tablas

Tabla 1 Requisitos Funcionales.....	22
Tabla 2 Actores del Sistema	29
Tabla 3 CU Gestionar Usuario	30
Tabla 4 Caso de Uso 3 <Gestionar Usuario>	32
Tabla 5 Descripción de componentes	43
Tabla 6 Diseño de casos de prueba Gestionar usuario.....	46
Tabla 7 Informe de la herramienta WebSecurity	50

Índice de Figuras

Figura 1 Modelo de Dominio	22
Figura 2 Diagrama de Casos de Uso del Sistema.....	31
Figura 3 Arquitectura de Drupal 7	34
Figura 4 Diagrama de clases del diseño	37
Figura 5 Diagrama de Clases de Diseño (Gestionar Usuario).....	38
Figura 6 Diagrama de secuencia. RF Insertar Usuario.....	38
Figura 7 Diagrama de secuencia. RF Modificar Usuario	39
Figura 8 Diagrama de secuencia. RF Eliminar Usuario	39
Figura 9 Diagrama de Despliegue.....	40
Figura 10 Diagrama de componentes	42
Figura 12 Resultado de las pruebas funcionales.....	49
Figura 13 Resultados de las pruebas de carga	51
Figura 14 Resultados de las pruebas de estrés	52

Introducción

En la era de la información, uno de los principales problemas es su exceso. Es necesario invertir mucho tiempo en ella debido, entre otras causas, a la liberación de los mecanismos regulatorios existentes en materia de publicaciones, sobre todo como resultado del surgimiento y desarrollo de Internet. Por esta razón, es frecuente encontrar un número significativo de publicaciones redundantes y de baja calidad mezcladas con otras importantes y sólidas, difíciles de hallar entre la información ruidosa que la "envuelve" (QUIROGA, 2002).

Con la creación de Internet, las empresas e instituciones vieron su surgimiento como una gran oportunidad para el intercambio de información y conocimiento. Además se eliminaba la limitación de sólo interactuar en un área geográfica en específico, ya que se puede navegar desde cualquier parte del mundo, hacia lugares muy distantes (PISCITELLI, 2005). Con este fin se crean los portales web, cuya función principal es: servir de puerta de entrada única para ofrecer al usuario, de forma fácil e integrada, el acceso a los recursos, contenidos y servicios que brinda.

En este contexto, las Tecnologías de la Información y las Comunicaciones (TIC) no son más que un medio para transmitir y gestionar datos e información. Con el desarrollo de estas tecnologías, ha surgido la oportunidad de encontrar nuevas maneras de llevar el quehacer de la sociedad a todos los rincones del mundo, adoptando el modelo de portal con el fin de resolver los problemas relacionados con la administración de la información y del conocimiento (ALMENARA, 2005).

Cuba no queda ajena a todas las transformaciones tecnológicas ocurridas a nivel internacional, pues en este momento está inmersa en un proceso de informatización de la sociedad, con el objetivo de integrarse plenamente a la infraestructura global de la información. Para darle cumplimiento a esta tarea ha creado instituciones que entre los servicios que brindan se encuentra el desarrollo de soluciones informáticas y portales web, un ejemplo de ello es la Universidad de las Ciencias Informáticas (UCI). En este centro existen varios sitios destinados a informar sobre el quehacer de la comunidad universitaria y de sus organizaciones, publicando informaciones, artículos, videos e imágenes que enriquecen los conocimientos y elevan el nivel cultural del personal, manteniéndolos actualizados del acontecer nacional e internacional.

En este centro de altos estudios conviven varias organizaciones políticas y de masas. Una de ellas es la Unión de Jóvenes Comunista (UJC), que agrupa en su seno a la juventud de vanguardia de la sociedad cubana y cuenta con un porcentaje elevado de miembros entre estudiantes, profesores y trabajadores.

Desde el propio surgimiento de la institución, las estructuras de la UJC han necesitado utilizar las potencialidades que las tecnologías de la información y las comunicaciones ofrecen, debido al gran número de actividades, informaciones e intercambio entre los miembros de la organización que conviven en la universidad. Por tanto resulta primordial, gestionar, divulgar y controlar el flujo de la información relacionada con el actuar de la organización y sus miembros.

En el año 2008 se realizó el lanzamiento de un portal para la UJC de la UCI con el propósito de informatizar la divulgación y promoción de sus actividades a través de la web. El sitio fue motivo de numerosas y fuertes críticas producto de la insatisfacción de los usuarios por la carencia de funcionalidades que permiten la gestión, modelación y aprobación de los contenidos, dificultando la actualización de los mismos. Además, se detectaron problemas que imposibilitaron el aprovechamiento de las posibilidades que brinda la web 2.0, el trabajo colaborativo, el intercambio, difusión de ideas, producir información, compartir en línea, entre otros (Rojas y otros, 2008).

Todo esto trajo consigo que hoy, el portal no se utilice y la divulgación sobre el accionar de la UJC en la UCI fuera realizada a través de las reuniones ordinarias o mediante el correo electrónico, lo que provoca que la información no se transmita de la misma manera a todos, dejando esta importante responsabilidad a la estructura de base. A su vez, se dificulta el intercambio directo de la base con el organismo superior, dependiendo de gestiones a terceros.

Todos estos elementos conllevan a que se plantee como **problema de investigación:** ¿Cómo contribuir desde la web a la divulgación y promoción del trabajo de la UJC en la Universidad de las Ciencias Informáticas?

Teniendo en cuenta lo antes expuesto el **objeto de estudio** se enmarca en la divulgación y promoción del trabajo de la UJC, delimitando así el **campo de acción** como la divulgación y promoción en la web del trabajo de la UJC en la UCI.

Proponiéndose como **objetivo general**, desarrollar un portal web para la UJC en la UCI que permita la divulgación y promoción del trabajo de la organización.

Objetivos específicos:

- 1- Caracterizar las tendencias actuales de los portales web para la difusión del trabajo de organizaciones políticas.
- 2- Diseñar un portal web para la UJC en la UCI teniendo en cuenta las funcionalidades y servicios con los que debe contar.
- 3- Implementar el portal web diseñado.
- 4- Validar la solución desarrollada mediante la realización de pruebas de software

Como **idea a defender** se plantea que la implementación del portal web para la UJC de la UCI permitirá contribuir a la divulgación y promoción del trabajo de la organización en esta institución.

Para dar cumplimiento a lo antes abordado se proponen una serie de **tareas** a desarrollar en el transcurso de la investigación:

- 1- Caracterización de las tendencias actuales de los portales web para la difusión del trabajo de organizaciones principalmente las políticas.
- 2- Identificación de requisitos funcionales y no funcionales para el desarrollo del portal web.
- 3- Validación de los requisitos aplicando las técnicas definidas.
- 4- Selección de las tecnologías, herramientas, estándares, patrones y metodologías que permitan desarrollar una solución.
- 5- Implementación de las funcionalidades del portal web a desarrollar.
- 6- Aplicación de pruebas de software a la funcionalidad a la solución desarrollada

Entre los **Métodos de trabajo científico** utilizados para darle cumplimiento a estas tareas se utilizaron:

Métodos teóricos:

Histórico-lógico: Empleado durante la investigación de los antecedentes y las tendencias actuales de los sistemas de gestión de información, además en la profundización de los conceptos y términos.

Analítico-sintético: Se utiliza en el estudio de las bibliografías para realizar la fundamentación teórica de la investigación y sintetizar los conceptos fundamentales que sean necesarios para la solución del problema.

Modelación: Se utilizó para modelar los principales procesos de negocio.

Método Empírico:

Entrevista: las entrevistas realizadas a dirigentes de la UJC posibilitaron la obtención de la información necesaria para realizar el diseño e implementación del software. Ver Anexo 1.

Observación: Este método es utilizado con la finalidad de caracterizar y analizar detalladamente cómo se realiza actualmente el proceso de gestión de información, identificando los mecanismos para desarrollar un sistema acorde con las necesidades.

Este documento está estructurado en introducción, tres capítulos, conclusiones, recomendaciones, referencia bibliográfica, bibliografía, anexos y glosario de términos.

En el capítulo I “Fundamentación Teórica del Portal de la UJC” se abordan los aspectos relacionados con los procesos de gestión de la información, además de la vinculación de estos con los portales web y la presentación de dicha información a través de este tipo de herramientas. Se describen además tecnologías y metodologías que se emplean para llegar a la solución.

El capítulo II “Características del sistema” se describe la solución propuesta partiendo del modelo de negocio. Se especifican los requerimientos funcionales y no funcionales del sistema, así como los casos de uso identificados.

En el capítulo III “Implementación y prueba del sistema” se describe la estructura del código implementado además de la presentación de los casos de prueba identificados para el sistema y los resultados de los mismos.

Capítulo 1. Fundamentación Teórica

En el presente capítulo se hace alusión a las características generales de un portal web. Se describen los conceptos fundamentales asociados al dominio del problema y el objeto de estudio, haciéndose un análisis de la situación actual. Se analizan las soluciones semejantes a la propuesta de solución; así como la fundamentación de la metodología, tecnologías y herramientas que se utilizarán para el desarrollo y construcción de la solución.

1.1 Conceptos asociados al dominio del problema

- **Militante:** miembro activo de una organización (UJC, 2014).
- **Comité UJC:** Estructura directiva principal que dirige a los Comités Primarios que se subordinan a esta o a Comités de Base (UJC, 2014).
- **Comité de Base:** Organización que reúne como mínimo a cinco militantes de la UJC, constituidos en los centros de trabajo, estudio, unidades militares, etc. para agrupar y organizar el trabajo de la organización. Es la unidad base de la estructura de la organización (UJC, 2014).
- **Comité Primario:** Órgano de dirección que dirige dos o más Comités de Base y se subordina al Comité UJC (UJC, 2014).
- **Actividad:** Conjunto de operaciones o tareas propias de una persona o entidad (DRAE, 1992).
- **Secretario General:** El secretario general como máximo representante de la organización de base, preside sus reuniones, controla diariamente las actividades de la UJC y adopta decisiones para enfrentar las situaciones cotidianas que se presentan en el centro, informándolas con posterioridad al comité para que sean ratificadas o no por este (UJC, 2014).

1.2 Historia y funcionamiento de la UJC

La UJC, surge como resultado del proceso de unidad del movimiento juvenil cubano, con el fin de movilizarlos en torno a la Revolución Socialista y contribuir a su educación política (UJC, 2014).

Organizada en todo el territorio nacional, con miembros de 16 hasta 35 años de edad, la UJC es cantera y reserva del Partido Comunista de Cuba (PCC), pues tiene entre sus principales responsabilidades la

formación ideológica en sus filas de los futuros militantes de la organización partidista, además de contribuir a la educación de las nuevas generaciones (UJC, 2014).

La UJC cuenta con un Buró Nacional, presidido por un primer y segundo secretario, un comité nacional, comités provinciales, municipales y locales en cada centro de trabajo o unidad militar (UJC, 2014).

1.3 Arquitectura de Información

Rosenfeld y Morville en el (2002) plantearon una visión más contemporánea del concepto de Arquitectura de Información (AI), enfocado como “El arte y la ciencia de estructurar y clasificar sitios web e intranets con el fin de ayudar a los usuarios a encontrar y manejar la información” (SÁNCHEZ, 2014).

Componentes de la Arquitectura de Información

La Arquitectura de la Información se puede separar en cuatro componentes: la organización, navegación, rotulado y sistemas de búsqueda. Cada uno de estos componentes cumple un papel fundamental en la arquitectura general del sitio web, y la deficiencia de uno de ellos puede ocasionar grandes problemas de usabilidad (SÁNCHEZ, 2014).

Beneficios que brinda la Arquitectura de la Información

- Generar estructuras que soporten el cambio y el crecimiento en el tiempo.
- Asegurar la consistencia y localización de la información.
- Crear sistemas de navegación intuitivos (SÁNCHEZ, 2014).

La AI es un elemento indispensable en cada una de las etapas de elaboración de aplicaciones web. El buen diseño de la AI, contribuye a la realización de un sistema accesible y usable, fortalece la fidelización del usuario con la aplicación y favorece la creación de un producto ampliamente aceptable.

1.4 Análisis de los sitios web homólogos

En el presente acápite se muestra un estudio sobre algunos portales web correspondientes a organizaciones o partidos juveniles dentro y fuera del país. Para la investigación se consultaron un total de 6 portales web.

1.4.1 Sistemas internacionales

Juventud Comunista del Ecuador:

La Juventud Comunista del Ecuador (JCE), es una organización política juvenil que se orienta por los principios del socialismo científico, del Marxismo-Leninismo y en la línea política y programa del Partido Comunista de Ecuador (PCE). Fue creada el 27 de agosto de 1929, naciendo inicialmente con el nombre de Federación Juvenil Comunista (ECUADOR, 2012).

El sitio fue creado en el año 2011. Este portal tiene como objetivo brindar información sobre el funcionamiento y estructura de los jóvenes en Ecuador. Entre sus principales deficiencias se encuentran la inexistencia de enlaces a redes sociales. No cuenta con un diseño adaptativo para dispositivos móviles y la página principal es muy extensa.

Juventud Comunista Paraguaya:

La Juventud Comunista Paraguaya (JCP) es un movimiento que agrupa a los militantes jóvenes del Partido Comunista Paraguayo (Paraguay, 2013).

Puede considerarse a la Federación Juvenil Comunista (FJC) de los años 30, como la precursora de la JCP. La FJC tenía un periódico llamado Patria Nueva, que se publicó durante la Primavera Democrática de 1946 (Paraguay, 2013).

Durante toda la segunda parte del siglo XX, la FJC no pudo organizar sus congresos ya que eran perseguidos por la policía de la dictadura. Recién, después de un trabajo de reorganización llevado a cabo por las nuevas generaciones del Partido Comunista Paraguayo, se pudo organizar el segundo congreso, en 2005, y se decide cambiar el nombre por el actual: Juventud Comunista Paraguaya (Paraguay, 2013).

El sitio fue creado en el año 2013. En este portal los usuarios tienen la posibilidad de realizar encuestas, conocer las noticias de última hora, galería de fotos y videos. Además brinda la posibilidad de acceder a diferentes redes sociales. El portal cuenta con un blog de noticias y opiniones.

Juventud Comunista de Venezuela:

La Juventud Comunista de Venezuela (JCV, se le suele llamar "jota-ce-ve") es una organización juvenil venezolana que agrupa a la juventud del Partido Comunista de Venezuela (PCV). Fueron fundadas el 16 de septiembre de 1947 como organización para la formación de cuadros para el PCV (Venezuela, 2013).

Se autodefine como una organización amplia, guiada por los principios del marxismo-leninismo, que basa su plataforma en la gesta de la independencia latinoamericana, de sus líderes Simón Bolívar y Antonio José de Sucre, así como en valores de la resistencia indígena de acuerdo con sus estatutos en principios internacionalistas y lucha por la unidad latinoamericana (Venezuela, 2013).

Forma parte de la Federación Mundial de la Juventud Democrática, de la cual es su organización coordinadora para América Latina y el Caribe (Venezuela, 2013).

En este portal los usuarios que lo visitan pueden ver una galería de imágenes y videos. En el menú tienen la posibilidad de seleccionar noticias de otros países que le interese al usuario, un buscador, resultados y estadísticas. Cuenta con un blog de noticias y debate, además de información sobre otros sectores como la economía y la cultura.

Unión de Juventudes Comunistas de España:

La Unión de Juventudes Comunistas de España (UJCE) o, simplemente, Juventud Comunista, es la organización juvenil del Partido Comunista de España (PCE). Actúa dentro de la línea estratégica y el programa del PCE, en cuya elaboración participa aportando la perspectiva juvenil. Se define como una organización política juvenil de clase, democrática, revolucionaria, republicana, laica e internacionalista. Su himno se llama "La Joven Guardia" (España, 2012).

Su origen se remonta a 1921, como resultado de la unión de las organizaciones juveniles comunistas vinculadas al Partido Comunista Español (fundado en 1920 por las Juventudes Socialistas) y al Partido Comunista Obrero Español (fundado en 1921 por militantes escindidos del PSOE). Actualmente está integrada en la Federación Mundial de la Juventud Democrática (España, 2012).

Este portal brinda la posibilidad de ver noticias de diferentes regiones y temas, cuenta con un buscador, muestra una galería de imágenes y de videos. Además permite el acceso a diferentes redes sociales y portales de interés del usuario.

Los cuatro tienen las siguientes características en común:

- Campo de búsqueda
- Galería de imágenes
- Galería de videos
- Noticias de última hora

- Entrevistas
- Menú con vínculo a diferentes redes sociales
- Encuestas
- Calendario de actividades de los jóvenes.

Estos portales no le dan solución al problema existente ya que fueron desarrollados para organizaciones privadas. No obstante, se pudo tomar algunas ideas de su estructura y los contenidos que publican, pudiendo ser útiles para el portal web a desarrollar.

1.4.2 Sistemas nacionales

En Cuba el desarrollo de la informática ha facilitado la gestión de la información en las diferentes empresas e instituciones y estas a su vez han podido mostrar su trabajo al mundo a través de Internet. Los portales web han constituido una vía para alcanzar estos fines. Son varios los ejemplos que se pueden citar en diversos ámbitos en nuestro país.

Diario de la Juventud Rebelde:

Juventud Rebelde es un periódico cubano, perteneciente a la Unión de Jóvenes Comunistas. El 21 de octubre de 1965, en el resumen de las actividades con motivo del V aniversario de la integración del Movimiento Juvenil Cubano y de la inauguración de los primeros Juegos Deportivos Nacionales, el Comandante en Jefe Fidel Castro Ruz anunció el surgimiento de un nuevo diario que tenía como antecedentes históricos a la revista Mella (semanario) y al Diario de la Tarde, y que debía recoger y continuar las tradiciones combativas y ejemplarizantes de la prensa juvenil cubana. (Rebelde, 2002)

Los enlaces son dirigidos por lo general a sitios que se relacionan con el portal visitado. Estos enlaces pueden residir en el pie de página del sitio o en cualquier menú. La publicación de noticias ha abierto una nueva puerta como medio de comunicación con carácter global, posibilitando mayores niveles de información y difusión que lo que permiten los medios tradicionales. Otra de las facilidades que brinda es el dinamismo en la actualización de las noticias, ya que se realiza de manera más fácil y eficiente. Existe seguimiento en redes sociales como *Facebook* o *Twitter*. Cuenta con un buscador.

En este portal los usuarios que lo visitan pueden ver una galería de imágenes y videos. En el menú tienen la posibilidad de seleccionar noticias nacionales e internacionales y de otros ámbitos como deportivas, culturales, ciencia y técnica que le interese al usuario.

Soy Cuba

Soy Cuba es un entorno digital de la Editora Juventud Rebelde que orienta sus intereses editoriales hacia la representación de la realidad juvenil y la satisfacción de necesidades interactivas, de facilitación social y entretenimiento, con énfasis en opciones para la participación y el diálogo *online* entre los jóvenes cubanos (Cuba, 2013).

El sitio busca generar mayores convergencias entre los espacios físicos de socialización juvenil y la comunicación virtual, desde una plataforma enfocada a compartir y potenciar el intercambio de experiencias contadas por los propios jóvenes. Sus prioridades no están en la difusión de noticias o informaciones, sino en una lectura a fondo de los procesos políticos, económicos, sociales y culturales que marcan la vida de esta generación (Cuba, 2013).

Concebido como un servicio público para y por los jóvenes cubanos, este entorno privilegia una filosofía colaborativa de trabajo que apuesta por el ejercicio del periodismo ciudadano joven; de modo que se construyan, de conjunto, mejores relatos, aproximaciones y lecturas críticas de la vida de los jóvenes, tomando en consideración las ventajas comunicativas de la web social (Cuba, 2013).

En este portal los usuarios pueden visitar una galería de imágenes y videos. Cuenta con noticias relacionadas a la política, los deportes, la cultura, la ciencia, la tecnología y la sociedad y con una sección de entretenimiento (juegos, postales, libros, recetas de cocina). Además un espacio dedicado a las opiniones y foros.

Sitio de la UJC en la UCI

En la universidad el sitio destinado a la UJC se encontraba con insuficiencias en la difusión de la información referente a temas de la organización, falta de interacción entre los usuarios y en lo relacionado con la inclusión de las nuevas informaciones. Estas carencias disminuyeron la motivación de los usuarios a su navegación y selección por lo que en estos momentos no se encuentra disponible para la comunidad universitaria. Se suma a ello, el no contarse con un interfaz de usuario amigable y amena (Rojas y otros, 2008).

Portal web de la Facultad 3 (Dragones)

El portal web de la Facultad 3 fue desarrollado mediante el Sistema Gestor de Contenidos Joomla en su versión 2.5. En este se puede encontrar una amplia gama de información del acontecer universitario,

nacional e internacional, proporcionando a los usuarios una vía para mantenerse constantemente informados. Facultad, Investigación y Postgrado, Avisos y Clasificados, y Noticias son algunas de las secciones que se pueden observar en el sitio. Además se puede incluir que:

- El portal no presenta elementos de interacción con las redes sociales.
- Se hace uso de elementos gráficos y de sonido.
- No brinda información de contacto.
- Su arquitectura pese a ser muy sencilla, posee los elementos fundamentales como la organización, navegación, rotulado y sistemas de búsqueda (Dragones, 2010).

A partir del análisis de homólogos realizado se propone incluir en el portal de la Unión de Jóvenes Comunistas (UJC) todos los elementos informativos, complementarios y de socialización abordados anteriormente y que el cliente apruebe, además de otros especificados por el mismo.

1.5 Sistema de Gestión de Contenidos (CMS)

Un CMS permite principalmente la creación y administración de contenidos de páginas web, así como controlar quién puede editar y visualizar los mismos. Estos sistemas poseen una o varias bases de datos donde se almacena el contenido del portal, lo que permite manejar los estilos o diseño de manera independiente. El uso de CMS permite:

1. Cambiar el diseño del portal web: Es muy fácil cambiar, gracias al sistema de plantillas que utiliza, las cuales se instalan y luego se seleccionan desde la interfaz administrativa.
2. Gestión de seguridad: Desde la interfaz administrativa también se controla completamente a los usuarios del portal, pudiendo especificar niveles jerárquicos, editar perfiles, censurar, dar permisos de publicación. Permite también enviar correos electrónicos masivos a todos los usuarios.
3. Administrar componentes: Es posible instalar, desinstalar y administrar componentes, por ejemplo galerías de fotos, sistemas de votaciones y publicación de noticias por parte de los usuarios.
4. Publicar contenidos: Es posible administrar contenidos que le permitirán formatear los textos con los estilos deseados en forma similar a un editor de texto como *Microsoft Word* (SHREVES, 2008).

CMS Joomla

Es un sistema de gestión de contenidos, y entre sus principales virtudes está la de permitir editar el contenido de un sitio web de manera sencilla. Es una aplicación de código.

Este gestor es relativamente flexible. Tiene una gran cantidad de módulos y extensiones para ampliar sus posibilidades de forma considerable. La comunidad de usuarios de este gestor está muy bien organizada y puede responder a la mayoría de incógnitas de sus participantes y esto se complementa con la ayuda documental que se encuentra en medios digitales como impresos (Santamaría, 2011).

CMS Drupal 7.26

Sistema de gestión de contenido 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 (Santamaría, 2011).

Este gestor es muy flexible. Tiene miles de extensiones para ampliar su funcionamiento. Tiene la capacidad de crear y modificar roles de usuarios, como también asignar privilegios a cada rol. La seguridad es uno de los aspectos mejor tratados con continuos parches, que con la buena configuración de los roles pueden hacer de este sistema la opción más segura. Uno de los puntos fuertes de Drupal es realizar filtros con los módulos *Views* y *Panels* para mostrar contenido. Existe una vasta documentación del sistema.

CMS WordPress

Wordpress es un Sistema de Gestión de Contenidos de gran sencillez de uso, por lo que es muy utilizado para la creación de Blog. Entre las funcionalidades que brinda el sistema está la de generar un archivo cronológico automáticamente, y posee un calendario, en los cuales es posible navegar y buscar información publicada en cualquier día, mes o año.

Wordpress presenta algunas desventajas, por ejemplo: no puede modificar el código de su sitio fácilmente. Dependiendo de la cantidad y frecuencia de las publicaciones la administración de Blog puede dificultarse, si no se hace revisión periódica. Por ser un espacio de acceso público pueden recibirse comentarios no deseados que no se encuentren relacionados con las temáticas. Si no se conoce con

certeza la dirección del Blog, la búsqueda se torna difícil. Carece de muchas de las funciones de redes sociales, comercio electrónico, foros, wikis y otras (Muras, 2009).

Selección del CMS

Teniendo en cuenta las características de estos CMS se puede concluir de que a pesar de que los dos son CMS de código abierto, o sea, que permiten modificar con libertad sus módulos es Drupal el que brinda mayores ventajas. Además, desde el punto de vista de la seguridad en cuanto al acceso a la web, permite diferenciar roles, asignando permisos específicos a cada uno, lo cual en *Joomla* queda un poco restringido por el uso de rangos y permisos específicos para los usuarios autenticados. *WordPress* no es el sistema adecuado para usar pues sus funcionalidades están encaminadas a la construcción de blogs, además de presentar varias desventajas, las cuales fueron expuestas anteriormente. Por estas razones se utilizará Drupal en su versión 7.26, la cual contiene mejoras en cuanto a rendimiento y seguridad; es mucho más flexible, más intuitiva y fácil de manejar.

1.6 Metodología de desarrollo

La metodología de desarrollo de *software* está definida como “*un conjunto de procedimientos, herramientas y un soporte documental que ayuda a los desarrolladores a realizar un nuevo software*” (Piattini, 1996).

Si se quiere construir un *software* de alta calidad, desarrollado en el tiempo planificado y con los costes establecidos, pero que además satisfaga la necesidad de ser elaborado de una forma más acelerada y que exista una reducción del costo del producto, es necesario enfocarse en trabajar de forma organizada, donde se controle y documente todo lo relacionado con el proyecto en cuestión y puedan eliminarse los riesgos que podrían presentarse durante el desarrollo del mismo, lo cual no podría lograrse sin el empleo de una metodología eficaz que se adapte a las características propias del *software* que se esté desarrollando.

Programación extrema (eXtreme Programming o XP)

XP está basada en la retroalimentación continua entre el cliente y el equipo de desarrollo, la comunicación fluida entre todos los participantes, la simplicidad en las soluciones implementadas y el coraje para enfrentar los cambios. Es especialmente adecuada para proyectos con requisitos imprecisos y muy

cambiantes, donde existe un alto riesgo técnico y el equipo de desarrollo es pequeño o mediano. Es una metodología altamente orientada a la implementación y genera poca documentación. Su ciclo de vida consiste de seis fases: Exploración, Planificación de la Entrega, Iteraciones, Producción, Mantenimiento y Muerte del proyecto. En todas las interacciones de este ciclo tanto el cliente como el programador aprenden, existiendo una retroalimentación entre ellos. Esta metodología utiliza las historias de usuario para especificar los requisitos del software, y el tratamiento de las mismas es muy dinámico y flexible. Además, cuenta con pocos roles entre los que se destacan el programador y el cliente (Amaro, y otros, 2007).

La metodología XP es la metodología ágil de más renombre en la actualidad y se diferencia de las demás metodologías de este tipo en el alto nivel de disciplina de las personas que participan en el proyecto (Amaro, y otros, 2007).

Proceso Unificado de Desarrollo (Rational Unified Process o RUP)

El Proceso Unificado de Desarrollo, es una metodología para el desarrollo de *software* orientado a objetos. Es un proceso de desarrollo de software, definido como un conjunto de actividades necesarias para transformar los requisitos de un usuario en un software. Está constituido por 5 flujos de trabajo fundamentales: requisitos, análisis, diseño, implementación y prueba, los cuales tienen lugar sobre 4 etapas o fases: inicio, elaboración, construcción y transición. Esta metodología es adaptable para proyectos a largo plazo y establece refinamientos sucesivos de una arquitectura ejecutable. RUP está basado en componentes y utiliza UML para visualizar, especificar y documentar cada una de las partes que comprenden el desarrollo de software (Heredia Ruiz, y otros, 2011).

Open Unified Process (OpenUp)

Es una versión más ágil de lo que es el *Rational Unified Process* (RUP), es un proceso unificado que aplica propuestas iterativas e incrementales dentro del ciclo de vida, tratando de ser manejable en relación con RUP. Plantea que se debe tener un software ya funcional, o lo que es lo mismo, un proyecto ejecutable en poco tiempo, para esto se debe utilizar sólo los procesos que sean necesarios, sin demasiados artefactos y sobre todo que el proyecto debe acoplarse a las necesidades del usuario, pudiendo ser éste modificado, mejorado y extendido.

OpenUP tiene dos ventajas importantes, este tipo de método disminuye los riesgos y además puede utilizarse tanto en proyectos pequeños como en proyectos grandes, aunque está concebida para proyectos pequeños. Si se maneja con cuidado y con profesionalismo se puede desarrollar un software de gran calidad, a pesar de que se le diseñe en poco tiempo y con poca documentación. *OpenUP* contiene las características esenciales de RUP, que incluye el desarrollo iterativo de casos de uso, además de proporcionar un acercamiento a la arquitectura central del sistema. El resultado es un proceso mucho más simple que sigue fielmente los principios de RUP.

Según la *Eclipse Foundation* (Foundation, 2012), *OpenUP* se organiza en dos dimensiones diferentes: Método y Proceso.

- Método: Los roles, las tareas y los artefactos están definidos, sin tener en cuenta cómo son aplicados en el ciclo de vida del proyecto.
- Proceso: Es cuando los elementos del proceso son aplicados en el sentido conductual, donde el mismo brinda los roles, las tareas y los artefactos. Pueden crearse ciclos de vida diferentes para proyectos diferentes.

OpenUP propone la generación de 13 artefactos fundamentales, los cuales permiten realizar una detallada descripción, organización y construcción de *software* por grupos pequeños de desarrollo. Los artefactos a generar (los 13 artefactos no son de uso obligatorio) dependen de las necesidades del equipo de desarrollo, el número de integrantes del mismo y los conocimientos que se tengan sobre el producto a desarrollar. La generación o no de un artefacto no incurre en la modificación de la metodología, *OpenUP* los propone y el equipo decide (Foundation, 2012).

Selección de la metodología de desarrollo de software

Después de analizar estas tres metodologías se ha decidido utilizar *OpenUP* porque preserva la esencia de RUP y al igual que él, posee un modelo de desarrollo iterativo e incremental, pero con la diferencia de que al ser un proceso mucho más ligero permite micro incrementos en breves períodos de tiempo, lo que posibilita ir creando liberaciones del producto mientras se desarrolla y efectuar modificaciones a los requerimientos sin altos costos en cuanto a tiempo, precio e implementación. Es apropiado para proyectos pequeños y de bajos recursos, permite detectar errores tempranos a través de un ciclo iterativo. Además,

evita la elaboración de documentación, diagramas e iteraciones innecesarias requeridas por RUP y puesto que es una metodología ágil, tiene un enfoque centrado en el cliente con iteraciones muy cortas.

1.7 Tecnologías y herramientas a utilizar en la implementación

1.7.1 Lenguajes de programación

Existe una multitud de lenguajes concebidos para la web. Cada uno de ellos explota más a fondo ciertas características que lo hacen más o menos útil para desarrollar distintas aplicaciones. Se pueden clasificar de dos formas diferentes; de acuerdo a la arquitectura Cliente/Servidor, los lenguajes del lado del servidor son aquellos reconocidos, ejecutados e interpretados por el propio servidor y que se envían al cliente en un formato comprensible para él; por otro lado, los lenguajes del lado del cliente son aquellos que pueden ser directamente comprendidos por el navegador y no necesitan un pre tratamiento. Cada uno de estos tipos tiene por supuesto sus ventajas y sus inconvenientes, por ejemplo: un lenguaje del lado del cliente es totalmente independiente del servidor, lo cual permite que la página pueda ser albergada en cualquier sitio. Inversamente, un lenguaje del lado servidor es independiente del cliente por lo que es mucho menos rígido respecto al cambio de un navegador a otro o respecto a las versiones del mismo.

Lenguaje del lado del servidor

Pre-Procesador de Hipertexto (PHP5.4)

PHP es un lenguaje dinámico muy usado en el desarrollo de aplicaciones web. El CMS Drupal está construido con este lenguaje lo cual proporciona la portabilidad de este. Es un lenguaje muy potente y estrechamente ligado a los desarrolladores webs por su versatilidad y muy usado en la construcción de webs dinámicas.

PHP es un lenguaje que desde sus inicios se hizo presente entre los desarrolladores web por sus facilidades en el procesamiento de información en formularios, manipulación de cookies y paginas dinámicas. Su soporte para bases de datos figura entre sus características más potentes. Integrándose también con otras bibliotecas externas le permite al desarrollador moverse con libertad permitiéndole a este desde generar documentos en PDF hasta analizar código XML (Henst, 2011).

Lenguaje del lado del cliente

Lenguaje marcado de hipertexto (HTML5). Constituye un conjunto de etiquetas utilizadas para definir una página web mediante su estructuración en forma de texto, imágenes, vínculos y otros elementos que la componen (Gómez, 2011).

JavaScript

Este lenguaje surge con la necesidad de lograr una web de mayor interacción con los usuarios, ya que el HTML solo permite crear páginas estáticas. *JavaScript* tiene amplias posibilidades ya que puede ser ejecutado del lado del cliente y también del lado del servidor (Oracle, 2014).

En Drupal es posible utilizar *JavaScript* de manera combinada con otros lenguajes enunciados anteriormente, esto hace posible que los sitios creados adquieran mayor dinamismo y resulten más atractivos por parte de los usuarios a los que están dirigidos.

Hojas de estilo de cascada (CSS)

Las hojas de estilo en cascada (*Cascading Style Sheets*), es un lenguaje usado para definir la presentación de un documento estructurado escrito en HTML. Constituye la mejor forma de separar los contenidos y su presentación, además es imprescindible para crear páginas web complejas (HECHEVARRIA, 2011).

Lenguaje de modelado

Lenguaje Unificado de Modelado (UML)

Lenguaje de Modelamiento Unificado (UML - *Unified Modeling Language*), se utiliza para especificar, visualizar, construir y documentar los artefactos de sistemas de *software*.

1.7.2 Sistema gestor de Base de Datos

Un Sistema de Gestión de Bases de Datos es aquel que se encarga de controlar el acceso concurrente, evitar redundancia, que se cumplan las restricciones y reglas de integridad, usar elementos que aceleren el acceso físico a los datos (índices, agrupamientos, funciones de dispersión), distribuir los bloques del

disco del modo más adecuado para el crecimiento y uso de los datos, controlar el acceso y los privilegios de los usuarios y recuperar ante fallos (BERNOT, 2012).

PostgreSQL 9.1

PostgreSQL es un Sistema de Gestor de Base de Datos (SGBD) que ofrece una potencia adicional al incorporar los conceptos de clases, herencia, tipos y funciones; de modo que los usuarios puedan entender fácilmente el sistema. *PostgreSQL* incluye reglas, restricciones, disparadores y una integridad transaccional que le aportan flexibilidad y potencia. Estas características incorporan a este gestor en la categoría de las bases de datos identificadas como objeto-relacionales, aunque este incluye algunos detalles pertenecientes al mundo de las bases de datos orientadas a objetos (Douglas, 2005).

Se escogió este Gestor de Base de Datos por la alta calidad que brinda, flexibilidad y potencia en su versión 9.1.

PgAdmin 1.14.0

Es una herramienta para la administración de la Base de Datos, de código abierto. Se encuentra disponible en más de una docena de lenguajes y para varios sistemas operativos. Esta herramienta simplifica en gran medida la administración.

1.7.3 Entorno integrado de desarrollo

NetBeans 7.1

La versión 7.1 de *NetBeans* dispone además de una integración avanzada con sistemas de control de versiones, posee un editor que integra los lenguajes tanto del lado del cliente como *JavaScript*, HTML, así como CSS y del lado del servidor PHP, brindando además todas las mejoras y ventajas que posee para PHP 5.3. Cuenta con un sistema de escaneo de todos los proyectos dentro del Entorno de Desarrollo Integrado (IDE), para reconocer funcionalidades, clases y proporcionar un completamiento de código eficiente de suma importancia para acelerar el desarrollo de las aplicaciones. Es un IDE de desarrollo maduro y con soporte a gran cantidad de librerías, lo cual lo convierte en un entorno de desarrollo ideal, por lo que permite a los programadores el uso de las mismas de una forma muy fácil e intuitiva. Es muy

sencillo de utilizar y exige muy poco conocimiento para su uso. Es completamente *software* libre y tiene versiones para los principales sistemas operativos como: *Windows*, *MacOS* y *LINUX* (Buschmann, 1996).

Por todas estas características se escoge para la realización de la aplicación web.

1.7.4 Servidor de aplicaciones web

Servidor Web Apache 2.2

Apache es una aplicación gratuita que convierte cualquier ordenador en un servidor web. Es de código abierto, flexible, rápido y eficiente. Apache permite negociar protocolos HTTP entre una máquina que haría de servidor web y los otros ordenadores que deseen ver determinado sitio web (Foundation, 2011).

Es multiplataforma y gracias a su popularidad es fácil encontrar documentación para profundizar en las funcionalidades que ofrece. Se caracteriza por su gran escalabilidad, seguridad y rendimiento. Trabaja con varios lenguajes como *Perl*, *Python* y PHP lo que permite desarrollar aplicaciones web de gran calidad (Foundation, 2011).

Se propone el uso de este servidor web porque es personalizable, la arquitectura modular de Apache permite construir un servidor hecho a la medida. Tiene la capacidad de correr en múltiples sistemas operativos y es de fácil adaptación a diferentes entornos y necesidades.

1.7.5 Herramienta Case

Visual Paradigm 8.0

Con el objetivo de mejorar los procesos de modelado, disminuyendo el tiempo de producción, facilitando la generación automática de la documentación y la reutilización de componentes, se ha determinado que la herramienta CASE a utilizar es *Visual Paradigm*. Otras características que ratifican el uso de la herramienta son su capacidad de trabajo multiplataforma y colaborativo, diseñado tanto para el sistema operativo *Windows* como para las diferentes distribuciones de *Linux*, facilitando la importación y exportación de componentes, ya sea como imagen o como archivos XML (Ltd, 2011).

Se elige esta herramienta ya que es multiplataforma y se puede utilizar tanto en *Linux* como en *Windows*. Su interfaz es muy intuitiva, de fácil aprendizaje para los desarrolladores. Permite la generación automática de diagramas a partir de descripciones de casos de usos.

1.7.6 Herramientas de pruebas de software

Apache JMeter

Apache *JMeter* es una herramienta de carga de código abierto, desarrollada en Java, diseñada para realizar pruebas de rendimiento y pruebas funcionales de software y aplicaciones web. Además permite simular una carga pesada en un servidor, una red o un objeto, como servidores web, de bases de datos, LDAP, de correos y otros, para poder comprobar su resistencia y analizar su rendimiento global en distintos tipos de carga (Caballero, 2012) .

Se puede utilizar para probar el rendimiento tanto estático como dinámico de los recursos (ficheros, objetos de Java, bases de datos, servidores de FTP y otros). Ofrece la posibilidad de visualizar varias estadísticas de carga y la reutilización de las pruebas en el desarrollo de otras nuevas (Martínez, 2009).

Se estará haciendo uso del Apache *JMeter* en su versión 2.3.1.

Acunetix

Acunetix Web Vulnerability Scanner es una potente utilidad que analiza una página web en busca de posibles vulnerabilidades. Detecta técnicas de hacking como, por ejemplo, inyección SQL, ataques de ejecución de código y ataques de autenticación, entre otros

Se estará haciendo uso de *Acunetix* en su versión 8.0.

1.8 Conclusiones Parciales

- A lo largo de todo el capítulo se caracterizaron los conceptos asociados al tema permitiendo una mayor comprensión de la problemática.
- El estudio de los sistemas similares tanto nacional como internacional permitió identificar los elementos que pueden ser utilizados para la implementación de la solución con el objetivo de satisfacer las necesidades del cliente.
- La definición de la metodología de desarrollo garantizó una guía del proceso a seguir, así como la documentación del proceso ingenieril. Por otra parte la definición de las tecnologías y herramientas permitirá construir un portal web acorde con los estándares internacionales.

Capítulo 2. Características del sistema

En este capítulo se abordan las características que el sistema a desarrollar debe tener, definiendo las clases del dominio y la relación entre ellas. Se identifican los requisitos funcionales y no funcionales que debe cumplir el portal web. Se explica además la arquitectura y el diseño del sistema a desarrollar.

2.1 Propuesta del sistema

El proceso que será informatizado es la difusión de la información referente a la organización de la Unión de Jóvenes Comunista. Para dar solución al problema existente se decide desarrollar un portal web utilizando el CMS *Drupal*, que podrá ser accedido por el usuario mediante una interfaz sencilla y amigable. Toda persona que desee obtener información referente a dicha organización puede visitar el sitio y encontrar en la página principal del mismo, un menú donde se mostrará información sobre la historia de la organización, reglamento de la misma, así como sesiones informativas en las cuales se publican las diferentes actividades a realizar, además de noticias nacionales e internacionales, encuestas, galerías de imágenes y videos.

El sistema propuesto tiene como objetivo brindar una plataforma de publicación de contenido, para obtener como resultado un portal donde situar de forma organizada, información sobre dicha organización.

2.2 Modelo de Dominio

El Modelo de Dominio representa los conceptos más importantes y significativos en el desarrollo de un sistema. Su objetivo fundamental es definir las interrelaciones de los objetos más importantes representados mediante clases. Es utilizado para la comprensión del entorno y en la planificación futura de la posible aplicación (ESPINOZA, 2010).

En el siguiente Modelo de Dominio se muestra cómo se maneja la información actualmente y se especifican las relaciones entre los principales conceptos que interactúan en el sistema:

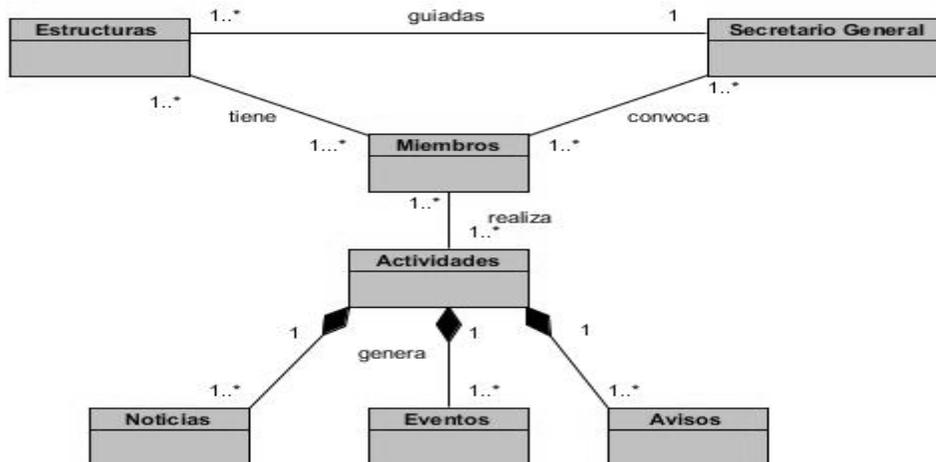


Figura 1 Modelo de Dominio

Estructuras: Se refiere a: Comité UJC, Comité Primario y Comité de Base.

2.3 Levantamiento de requisitos

Como paso fundamental en el proceso de desarrollo de la solución propuesta se realizará el levantamiento de requisitos. Existen dos tipos de requisitos, los funcionales que son las condiciones o capacidades que un producto debe cumplir y los no funcionales que representan las propiedades o cualidades que el producto debe tener (Linares, 2010).

2.3.1 Requisitos Funcionales

Los requisitos funcionales están referidos a las funciones que debe cumplir el sistema y pueden verse como capacidades o condiciones que el sistema debe poseer. Acto seguido se enumeran los requisitos funcionales para la confección del portal web (Linares, 2010).

Para ver la descripción, prioridad y complejidad de los requisitos funcionales obtenidos dirigirse al Anexo 9.

Tabla 1 Requisitos Funcionales

No	Nombre	Descripción	Prioridad para el cliente	Complejidad

RF1	Insertar Usuario	El sistema debe permitir insertar un usuario.	Alta	Baja
RF2	Modificar Usuario	El sistema debe permitir modificar un usuario.	Alta	Baja
RF3	Eliminar Usuario	El sistema debe permitir eliminar un usuario.	Alta	Baja
RF4	Mostrar Usuarios	El sistema debe permitir mostrar los datos de un usuario.	Alta	Baja
RF5	Insertar Sección	El sistema debe permitir insertar temáticas.	Alta	Media
RF6	Modificar Sección	El sistema debe permitir editar las temáticas.	Alta	Media
RF7	Eliminar Sección	El sistema debe permitir eliminar temáticas.	Alta	Media
RF8	Mostrar Secciones	El sistema debe permitir mostrar las temáticas.	Alta	Media
RF9	Insertar Noticia	El sistema debe permitir insertar una Noticias	Alta	Alta
RF10	Modificar Noticia	El sistema debe permitir editar una Noticias	Media	Media
RF11	Eliminar Noticia	El sistema debe permitir eliminar una Noticias	Alta	Alta
RF12	Mostrar Noticias	El sistema debe permitir mostrar las últimas Noticias en la página principal.	Media	Media
RF13	Votar Contenido	El sistema debe permitir votar por el mejor contenido.	Media	Media

RF14	Mostrar Contenidos más Populares	El sistema debe mostrar los contenidos más populares.	Media	Media
RF15	Insertar Aviso	El sistema debe permitir insertar un Aviso.	Alta	Media
RF16	Modificar Aviso	El sistema debe permitir editar un Aviso.	Alta	Media
RF17	Eliminar Aviso	El sistema debe permitir eliminar un Aviso.	Alta	Media
RF18	Mostrar Avisos	El sistema debe permitir mostrar los Aviso en la página principal.	Alta	Media
RF19	Insertar Documento	El sistema debe permitir insertar un Documento.	Alta	Media
RF20	Modificar Documento	El sistema debe permitir editar un Documento.	Alta	Media
RF21	Eliminar Documento	El sistema debe permitir eliminar un Documento.	Alta	Media
RF22	Mostrar Documento	El sistema debe permitir mostrar los Documento.	Alta	Media
RF23	Insertar Galería	El sistema debe permitir insertar fotos en la galería.	Media	Media
RF24	Modificar Galería	El sistema debe permitir editar fotos en la galería.	Media	Media
RF25	Eliminar Galería	El sistema debe eliminar fotos en la galería.	Media	Media
RF26	Mostrar Galerías	El sistema debe permitir mostrar fotos en la galería.	Media	Media
RF27	Insertar Evento	El sistema debe permitir insertar eventos	Media	Media

RF28	Modificar Evento	El sistema debe permitir editar los eventos	Media	Media
RF29	Eliminar Evento	El sistema debe eliminar los eventos	Media	Media
RF30	Mostrar Eventos	El sistema debe permitir mostrar los eventos	Media	Media
RF31	Insertar Frente de Trabajo	El sistema debe permitir insertar Frente de Trabajo.	Media	Media
RF32	Modificar Frente de Trabajo	El sistema debe permitir editar Frente de Trabajo.	Media	Media
RF33	Eliminar Frente de Trabajo	El sistema debe eliminar Frente de Trabajo.	Media	Media
RF34	Mostrar Frentes de Trabajo	El sistema debe permitir mostrar Frente de Trabajo.	Media	Media
RF35	Insertar Información Mensual	El sistema debe permitir insertar Información Mensual.	Media	Media
RF36	Modificar Información Mensual	El sistema debe permitir editar Información Mensual.	Media	Media
RF37	Eliminar Información Mensual	El sistema debe eliminar Información Mensual.	Media	Media
RF38	Mostrar Informaciones Mensuales	El sistema debe permitir mostrar Información Mensual.	Media	Media
RF39	Insertar Video	El sistema debe permitir insertar Video	Media	Media

	Promocional	Promocional.		
RF40	Modificar Video Promocional	El sistema debe permitir editar Video Promocional.	Media	Media
RF41	Eliminar Video Promocional	El sistema debe eliminar Video Promocional.	Media	Media
RF42	Mostrar Videos Promocionales	El sistema debe permitir mostrar Video Promocional.	Media	Media
RF43	Insertar Encuesta	El sistema debe permitir insertar una Encuesta.	Media	Media
RF44	Modificar Encuesta	El sistema debe permitir editar una Encuesta.	Media	Media
RF45	Eliminar Encuesta	El sistema debe eliminar una Encuesta.	Media	Media
RF46	Mostrar Encuestas	El sistema debe permitir mostrar Encuestas.	Media	Media
RF47	Insertar Comentario	El sistema debe permitir insertar un Comentario.	Media	Media
RF48	Modificar Comentario	El sistema debe permitir editar un Comentario.	Media	Media
RF49	Eliminar Comentario	El sistema debe eliminar un Comentario.	Media	Media
RF50	Mostrar Comentarios	El sistema debe permitir mostrar Comentarios.	Media	Media
RF51	Insertar Foro	El sistema debe permitir insertar contenidos del tipo Tópicos en el foro	Alta	Media
RF52	Modificar Foro	El sistema debe permitir modificar los contenidos del tipo Tópicos en el foro	Alta	Media

RF53	Eliminar Foro	El sistema debe permitir eliminar los contenidos del tipo Tópicos en el foro	Alta	Media
RF54	Búsqueda Simple	El sistema debe permitir realizar búsquedas simples sobre las publicaciones.	Alta	Alta
RF55	Búsqueda Avanzada	El sistema debe permitir realizar búsquedas avanzadas sobre las publicaciones	Alta	Alta
RF56	Autenticar Usuario	El sistema debe permitir al usuario autenticarse, introduciendo su nombre de usuario y contraseña.	Alta	Baja

2.3.2 Requisitos no Funcionales

Los requisitos no funcionales tienen que ver con características que de una u otra forma puedan limitar el sistema, como por ejemplo, el rendimiento (en tiempo y espacio), interfaces de usuario, fiabilidad (robustez del sistema, disponibilidad de equipo), mantenimiento, seguridad, portabilidad, estándares, etc. Los requisitos no funcionales están enfocados a elementos que el sistema debe tener en cuenta a la hora de su confección, expresan cualidades o propiedades necesarias para lograr un mejor funcionamiento y aceptación por parte del cliente. A continuación se describen los requisitos no funcionales a tener en cuenta en el desarrollo del portal web (Oliveros, 2012).

Usabilidad

RnF1 Diseño limpio y claro que facilite la navegación.

RnF2 Se debe lograr una estructura simple y lo más intuitiva posible de la arquitectura de la información.

Seguridad

RnF3 Se asignarán los permisos de acceso, escritura, lectura en dependencia del rol que desempeñe cada usuario del sistema.

RnF4 El sistema debe permitir usuarios con diferentes niveles de acceso.

RnF5 La información manejada por el sistema estará protegida de acceso no autorizado y divulgación.

RnF6 Llevar un registro de sucesos donde se archiven los eventos del sistema incluyendo los eventos de error, inicio de sesión, cierre de sesión y modificación de la información.

RnF7 En caso de que el sistema presente alguna falla, los errores se deben mostrar sin detalles de información que pueda comprometer la seguridad e integridad del mismo.

RnF8 La información manejada por el sistema será objeto de cuidadosa protección contra la corrupción y estados inconsistentes.

Soporte

RnF9 El sistema debe dar la posibilidad de ser mejorado, así como de incorporarle nuevos servicios en caso de ser necesarios.

Diseño

RnF10 Las páginas generadas deben cumplir con los estándares para XHTML 1.0 y CSS 2.0.

RnF11 El software, bibliotecas o componentes empleados deben estar regidos en la medida de lo posible por licencias y patentes de software libre.

RnF12 Por parte del cliente se requiere un navegador capaz de interpretar *JavaScript*.

Accesibilidad

RnF13 Interfaces fáciles de usar por personas que tienen poca experiencia en el manejo de computadoras.

Disponibilidad

RnF14 Se garantizará la disponibilidad de la información en el sistema durante todo el día, y que los dispositivos o mecanismos empleados para lograr la seguridad no retrasen la obtención de los datos por parte de los usuarios en un momento dado.

Interfaz

RnF15 La aplicación debe ser visible en navegadores como Internet *Explorer* 6 o superior, *Mozilla Firefox*, *Chrome* y *Opera*.

RnF16 La comunicación entre el cliente y el servidor web será realizada a través del protocolo HTTPS por el puerto 443.

RnF17 La comunicación entre el servidor web y el servidor de Base de Datos será realizada través del protocolo TCP: 5432.

Requisitos de Licencia

RnF18 Uso de la licencia PHP *License*.

RnF19 Uso de la licencia *Apache Software*.

RnF20 Uso de la licencia BSD de *PosgreSQL*.

RnF21 Uso de la licencia GNU/GPL para el CMS *Drupal*.

Estándares Aplicables

RnF22 Hojas de estilos en cascada (CSS).

RnF23 Se siguen el estándar de programación sugerido por los desarrolladores del CMS *Drupal*.

Hardware

RnF 24 Debe existir una red de área local para la divulgación de la información.

RnF 25 El servidor de aplicaciones web y de Base de Datos deben poseer como mínimo un CPU Core 2Duo a 2.20GHz.

RnF 26 El servidor de aplicaciones web debe poseer una capacidad mínima de 120 GB.

RnF 27 El servidor de Base de datos debe poseer una capacidad mínima de 320 GB.

RnF 28 Los servidores web y de Base de Datos deben poseer como mínimo 1 GB de memoria RAM.

2.4 Definición de los Casos de Uso del Sistema

2.4.1 Actores del sistema

Los usuarios relacionados con el sistema son todos aquellos que de una forma u otra van a interactuar con la aplicación, incluyendo a los que mantienen el sistema actualizado en correcto funcionamiento y a los que utilizan sus servicios.

Tabla 2 Actores del Sistema

Nombre del Actor	Justificación
Administrador	Persona encargada de dar soporte al sistema, administrar los usuarios del

	sistema.
Editor	Es la persona encargada de editar el portal, es decir, crear, eliminar y actualizar el contenido, tanto de noticias, artículos, avisos u otras informaciones como de los diferentes eventos que se llevarán a cabo en el portal.
Usuario Anónimo	Son los estudiantes, profesores, trabajadores u otras personas que hagan uso del sistema, realizando operaciones de acceso sobre el mismo, sin que requiera ser autenticado. Como usuario puede participar del foro, consultar los documentos, e informarse sobre las actividades y eventos a realizarse por la organización.

2.4.2 Casos de uso del sistema

A continuación se ofrece algunos de los casos de uso del portal. Todos ellos son inicializados por el actor. Los restantes se pueden encontrar en los anexos. Ver Anexo 2

Tabla 3 CU Gestionar Usuario

Caso de Uso 3	Gestionar Usuario
Actor	Administrador
Descripción	El administrador accede al sistema para otorgarle los diferentes permisos a los usuarios según sus roles y define la

	configuración del portal.
Referencia	RF1, RF2, RF3, RF4

2.4.3 Diagrama de Casos de Uso del Sistema

Los casos de uso representan procesos o funcionalidades que se realizan en el sistema, constituyen una descripción de cómo actúa el sistema. En la metodología OpenUP este artefacto presenta una visión general del comportamiento previsto del sistema. Es la base para un acuerdo entre las partes interesadas y el equipo del proyecto en lo que respecta a la funcionalidad prevista del sistema. También guía a las distintas tareas del ciclo de vida de desarrollo de software.

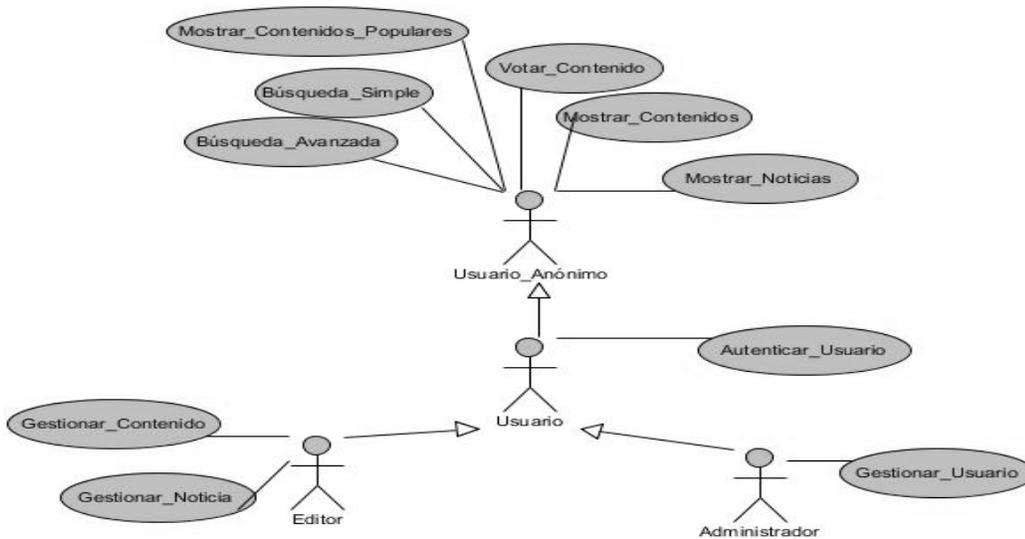


Figura 2 Diagrama de Casos de Uso del Sistema

2.4.4 Descripción de Casos de Uso del Sistema

A continuación se ofrece la descripción del caso de uso Gestionar Usuario. Los restantes casos de uso se pueden encontrar en los Anexos. Ver Anexo 3

Tabla 4 Caso de Uso 3 <Gestionar Usuario>

Caso de Uso																																																													
Caso de Uso 3	Gestionar Usuario																																																												
Propósito	Crear, modificar, eliminar y modificar usuario.																																																												
Actores	Administrador																																																												
Resumen	El Caso de Uso inicia cuando el administrador accede al sistema para otorgarle diferentes permisos a los usuarios según sus roles.																																																												
Complejidad	Alta																																																												
Prioridad	Alta																																																												
Precondiciones	Que el administrador esté previamente autenticado																																																												
Poscondiciones	Que se haya creado, modificado, eliminado los usuarios.																																																												
Prototipo																																																													
<p>+ Añadir usuario</p> <div style="border: 1px solid #ccc; padding: 5px;"> <p>MOstrar SÓLO LOS USUARIOS EN LOS QUE</p> <table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 15%;">rol</td> <td style="width: 35%;"> <input type="text" value="cualquiera"/> </td> <td style="width: 10%; text-align: right;">Filtro</td> </tr> <tr> <td>permiso</td> <td> <input type="text" value="cualquiera"/> </td> <td></td> </tr> <tr> <td>estado</td> <td> <input type="text" value="cualquiera"/> </td> <td></td> </tr> </table> </div> <div style="border: 1px solid #ccc; padding: 5px; margin-top: 5px;"> <p>ACTUALIZAR OPCIONES</p> <table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 80%;"> <input type="text" value="Desbloquear a los usuarios seleccionados"/> </td> <td style="width: 20%; text-align: right;">Actualizar</td> </tr> </table> </div> <table border="1" style="width: 100%; border-collapse: collapse; margin-top: 5px;"> <thead> <tr> <th style="width: 5%;"></th> <th style="width: 20%;">NOMBRE DE USUARIO</th> <th style="width: 10%;">ESTADO</th> <th style="width: 15%;">ROLES</th> <th style="width: 15%;">MIEMBRO DURANTE</th> <th style="width: 10%;">ÚLTIMO ACCESO</th> <th style="width: 25%;">OPERACIONES</th> </tr> </thead> <tbody> <tr> <td><input type="checkbox"/></td> <td>admin</td> <td>activo</td> <td>• administrator</td> <td>3 meses 1 semana</td> <td>hace 1 min 58 segs</td> <td>editar</td> </tr> <tr> <td><input type="checkbox"/></td> <td>usuario</td> <td>activo</td> <td></td> <td>3 meses 1 semana</td> <td>hace 27 mins 20 segs</td> <td>editar</td> </tr> <tr> <td><input type="checkbox"/></td> <td>editor1</td> <td>activo</td> <td>• Editor</td> <td>1 semana 4 días</td> <td>hace 37 mins 13 segs</td> <td>editar</td> </tr> <tr> <td><input type="checkbox"/></td> <td>yeni</td> <td>bloqueado</td> <td></td> <td>49 mins 39 segs</td> <td>nunca</td> <td>editar</td> </tr> <tr> <td><input type="checkbox"/></td> <td>editor</td> <td>activo</td> <td></td> <td>32 mins 2 segs</td> <td>hace 23 mins 14 segs</td> <td>editar</td> </tr> <tr> <td><input checked="" type="checkbox"/></td> <td>ed</td> <td>activo</td> <td>• Editor</td> <td>1 min 17 segs</td> <td>nunca</td> <td>editar</td> </tr> </tbody> </table>		rol	<input type="text" value="cualquiera"/>	Filtro	permiso	<input type="text" value="cualquiera"/>		estado	<input type="text" value="cualquiera"/>		<input type="text" value="Desbloquear a los usuarios seleccionados"/>	Actualizar		NOMBRE DE USUARIO	ESTADO	ROLES	MIEMBRO DURANTE	ÚLTIMO ACCESO	OPERACIONES	<input type="checkbox"/>	admin	activo	• administrator	3 meses 1 semana	hace 1 min 58 segs	editar	<input type="checkbox"/>	usuario	activo		3 meses 1 semana	hace 27 mins 20 segs	editar	<input type="checkbox"/>	editor1	activo	• Editor	1 semana 4 días	hace 37 mins 13 segs	editar	<input type="checkbox"/>	yeni	bloqueado		49 mins 39 segs	nunca	editar	<input type="checkbox"/>	editor	activo		32 mins 2 segs	hace 23 mins 14 segs	editar	<input checked="" type="checkbox"/>	ed	activo	• Editor	1 min 17 segs	nunca	editar
rol	<input type="text" value="cualquiera"/>	Filtro																																																											
permiso	<input type="text" value="cualquiera"/>																																																												
estado	<input type="text" value="cualquiera"/>																																																												
<input type="text" value="Desbloquear a los usuarios seleccionados"/>	Actualizar																																																												
	NOMBRE DE USUARIO	ESTADO	ROLES	MIEMBRO DURANTE	ÚLTIMO ACCESO	OPERACIONES																																																							
<input type="checkbox"/>	admin	activo	• administrator	3 meses 1 semana	hace 1 min 58 segs	editar																																																							
<input type="checkbox"/>	usuario	activo		3 meses 1 semana	hace 27 mins 20 segs	editar																																																							
<input type="checkbox"/>	editor1	activo	• Editor	1 semana 4 días	hace 37 mins 13 segs	editar																																																							
<input type="checkbox"/>	yeni	bloqueado		49 mins 39 segs	nunca	editar																																																							
<input type="checkbox"/>	editor	activo		32 mins 2 segs	hace 23 mins 14 segs	editar																																																							
<input checked="" type="checkbox"/>	ed	activo	• Editor	1 min 17 segs	nunca	editar																																																							
Flujo básico																																																													
Acción del Actor	Respuesta del Sistema																																																												
1. El administrador accede al sistema para	1.1 El sistema muestra los campos que se																																																												

agregar un nuevo usuario.	deben llenar para agregar al usuario.
2. El administrador llena todos campos mostrados.	2.1 El sistema verifica que todos los campos hayan sido llenados correctamente. 2.2 El sistema crea el nuevo usuario.
Flujo alternativo	
Sección "Modificar Usuario"	
Acción del actor	Respuesta del Sistema
3. El administrador el usuario que desea modificar.	3.1 El sistema muestra una lista de datos dando la posibilidad de modificarlos.
4. El administrador modifica los datos deseados y los envía.	4.1 El sistema verifica que todos los datos estén correctos y los obligatorios estén llenos. 4.2 El sistema actualiza los nuevos datos y finaliza el caso de uso.
Sección "Eliminar Usuario"	
5. El administrador selecciona al usuario que desea eliminar.	5.1 El sistema pregunta para confirmar y elimina el usuario.
Sección "Mostrar Usuarios"	
5. El administrador selecciona los usuarios autenticados.	5.1 El sistema muestra todos los usuarios autenticados.

2.5 Arquitectura de Drupal

El sistema está implementado sobre Drupal, por lo cual la arquitectura se hereda de este CMS.

La arquitectura de software es una vista del sistema que incluye los componentes principales del mismo, la conducta de esos componentes según se percibe desde el resto del sistema y las formas en que los componentes interactúan y se coordinan para alcanzar la misión del sistema (Clements).

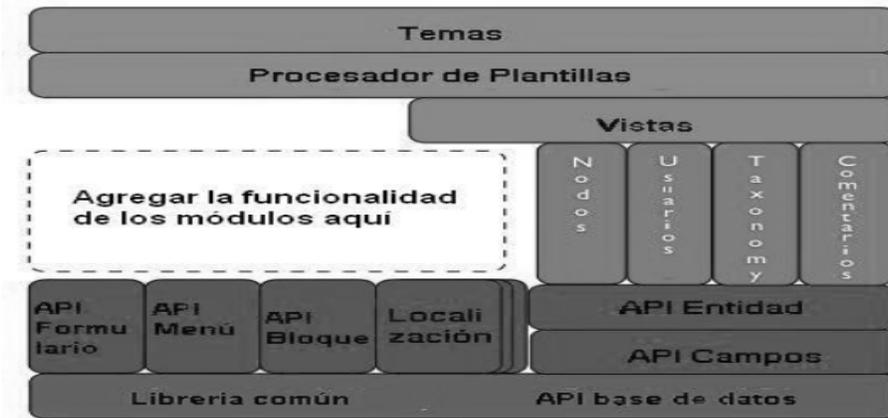


Figura 3 Arquitectura de Drupal 7

CMS Drupal en su versión 7, muestra una arquitectura de 5 capas, siendo la primera la responsable de interactuar con los usuarios a través del Tema desarrollado para dar solución al problema planteado, en esta capa se encuentra también el procesador de plantillas.

La segunda capa es la encargada de manejar las vistas del sitio, muestran en los temas los cambios realizados a través de los módulos, nodos, taxonomías, usuarios y comentarios, que serían una tercera capa, por lo que las vistas son las encargadas de crear un enlace entre la capa uno y la capa tres.

En la cuarta capa se encuentran los formularios, los menús, bloques, entidades, y campos. En la quinta capa, se encuentran las bibliotecas, y la Base de Datos, fuente de procedencia de la mayor parte de la información que se muestra en el sitio, en su conjunto todos potencian la escalabilidad del CMS.

2.6 Patrones de diseño

“Los patrones de diseño de *software* son soluciones reutilizables de problemas recurrentes que aparecen durante el proceso de diseño de *software* orientado a objetos” (LARMAN, 2002).

Durante el desarrollo del sistema se usaron los siguientes patrones:

Patrones GRASP

Los Patrones Generales de Software para Asignar Responsabilidades (GRASP, del inglés *General Responsibility Assignment Software Patterns*) permiten, como su nombre lo indica, la asignación de

responsabilidades a objetos. Aunque se considera que más que patrones propiamente dichos, son una serie de "buenas prácticas" de aplicación recomendable en el diseño de software (LARMAN, 2002).

- **Bajo Acoplamiento:** La utilización de este patrón permite tener las clases lo menos ligadas entre sí que se pueda. De tal forma que en caso de producirse una modificación en alguna de ellas, se tenga la mínima repercusión posible en el resto de clases, potenciando la reutilización, y disminuyendo la dependencia entre las clases. Este patrón se utiliza en todas las clases del sistema.
- **Alta cohesión:** Define que la información almacenada en una clase debe ser coherente y estar relacionada con esta. Propone además, que no se debe saturar una clase de métodos, sino asignar las responsabilidades a cada clase correspondiendo a la información que almacena. Este patrón se utiliza en todas las clases del sistema.

Patrones GoF

Los patrones GoF (La Pandilla de los Cuatro, del inglés *Gang-Of-Four*) son en el campo del diseño orientado a objetos los más conocidos y usados en la actualidad. Estos describen las formas en las que pueden ser organizados los objetos para trabajar unos con otros (Gamma, 2005).

- **Observador (*Observer*):** Define una dependencia entre objetos, de forma tal que cuando uno de los objetos cambie su estado el observador notifica este cambio y se actualizan todos los objetos que dependen de él. El objetivo principal es desacoplar la clase de los objetos clientes del objeto aumentando la modularidad del lenguaje.

El patrón Observador es generalizado en Drupal. Cuando una modificación es hecha a un vocabulario en el sistema de taxonomía de Drupal, el *hook taxonomy* es llamado en todos los módulos que lo implementan. Mediante la aplicación del *hook* (ganchos), se han registrado como observadores del objeto vocabulario; cualquier cambio a esto puede entonces actuar como es apropiado.

- **Puente (*Bridge*):** Desacopla una abstracción de su implementación, de modo que ambas puedan ser variadas de forma independiente, sin modificar una a la otra.

La capa de abstracción de la bases de datos de Drupal es similar al patrón de diseño bridge. Los módulos necesitan ser escritos de forma tal que sean independientes del gestor de bases de datos

que se esté usando, y proporciona la capa de abstracción para ello. Permite añadir soporte para otros gestores de bases de datos sin la necesidad de modificar el código del módulo.

- **Cadena de Responsabilidad (*Chain of Responsibility*):** Evita acoplar el emisor de una petición a su receptor, dando a más de un objeto la posibilidad de responder a la petición. Permite establecer una cadena de los objetos receptores a través de los cuales pasan una petición formulada por algún objeto emisor.

El sistema de menús de Drupal sigue el patrón Cadena de responsabilidades. En cada solicitud de la página, el menú del sistema determina: si hay un módulo para gestionar la solicitud, si el usuario tiene acceso a los recursos solicitados, y que la función se llama para hacer el trabajo. Para ello, el mensaje se pasa a la opción del menú correspondiente a la vía de la solicitud. Si el elemento de menú no puede manejar la petición, se pasa de la cadena. Esto continúa hasta que un módulo se encarga de la petición, un módulo niega el acceso para el usuario, o la cadena se ha agotado.

- **Patrón de reflexión (*Reflection Pattern*):** Describe el comportamiento de todo sitio basado en Drupal. Gracias a este patrón el sistema se convierte en una aplicación muy adaptable a diferentes entornos puesto que permite prácticamente la modificación de todo su comportamiento a través de los módulos instalables, sin la necesidad de modificar el núcleo.

2.7 Diagrama de Clases del Diseño

El Diagrama de Clases es el diagrama principal de diseño y análisis para un sistema. Permite visualizar los paquetes que genera el CMS. A continuación se realiza una breve descripción de los paquetes generados por Drupal (Gil, 2012).

“Includes”: Contiene un conjunto de ficheros indispensables para el correcto funcionamiento del CMS, por ejemplo los ficheros de conexión a la Base de Datos.

“Themes”: Incluye las distintas plantillas que serán utilizadas como tema para la interfaz de Drupal.

“Modules”: Se encuentran ubicados los módulos que permiten el funcionamiento del CMS.

“Scripts”: Almacena los ficheros que permiten que sean visualizados correctamente los datos (Gil, 2012).

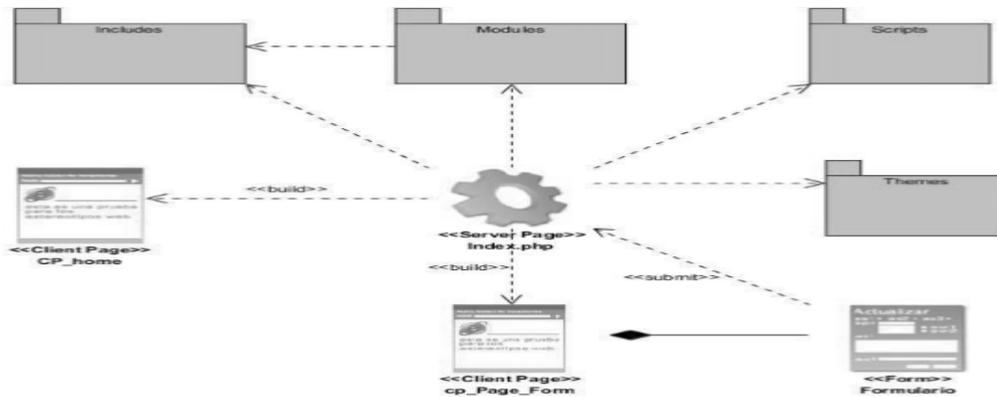


Figura 4 Diagrama de clases del diseño

Una vez confeccionado el diagrama de clases del diseño por paquetes, se modela el diagrama de clases del diseño para cada subpaquete como se muestra a continuación. Los restantes diagramas asociados a la solución propuesta son incluidos en los anexos del presente trabajo. Ver Anexo 4.

Paquete Gestionar Usuario

Para insertar un usuario la página servidora SP_GestionarUsuario construye la página cliente CP_InsertarUsuario, la última se compone de un formulario Form_InsertarUsuario donde se van a introducir los datos y que va a ser ejecutado por la página servidora. Para modificar un usuario la página servidora construye la página cliente CP_Modificar usuario, la última se compone de un formulario Form_ModificarUsuario donde se van a modificar los datos y que va a ser ejecutado por la página servidora. Para eliminar el usuario la página servidora construye la página cliente CP_EliminarUsuario, la última se compone de un formulario Form_EliminarUsuario donde se van a eliminar los datos y que va a ser ejecutado por la página servidora. Para mostrar un usuario la página servidora construye la página cliente CP_MostrarUsuario, la última se compone de un formulario Form_MostrarUsuario donde se van a mostrar los datos.

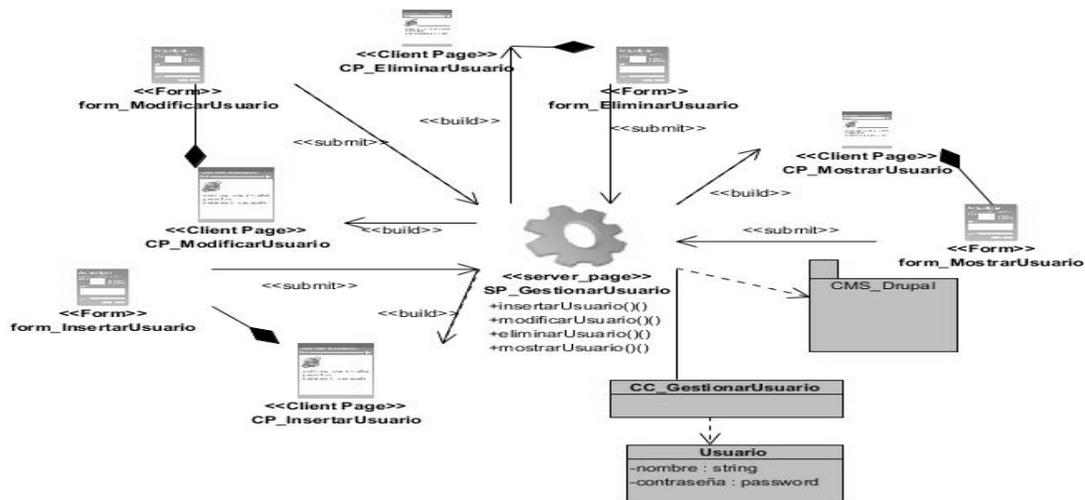


Figura 5 Diagrama de Clases de Diseño (Gestionar Usuario)

2.8 Diagrama de Secuencia

Un diagrama de secuencia muestra las interacciones entre objetos, ordenadas en secuencia temporal durante un escenario concreto. A continuación se representan los diagramas de secuencia de los requisitos 1, 2 y 3. Los restantes diagramas de secuencia asociados a la solución propuesta son incluidos en los anexos del presente trabajo. Ver Anexo 5.

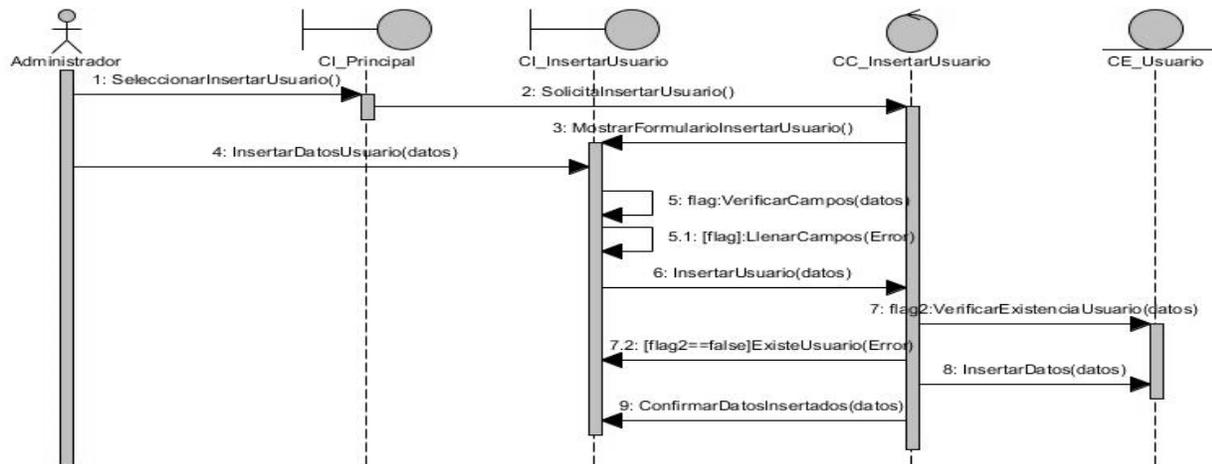


Figura 6 Diagrama de secuencia. RF Insertar Usuario

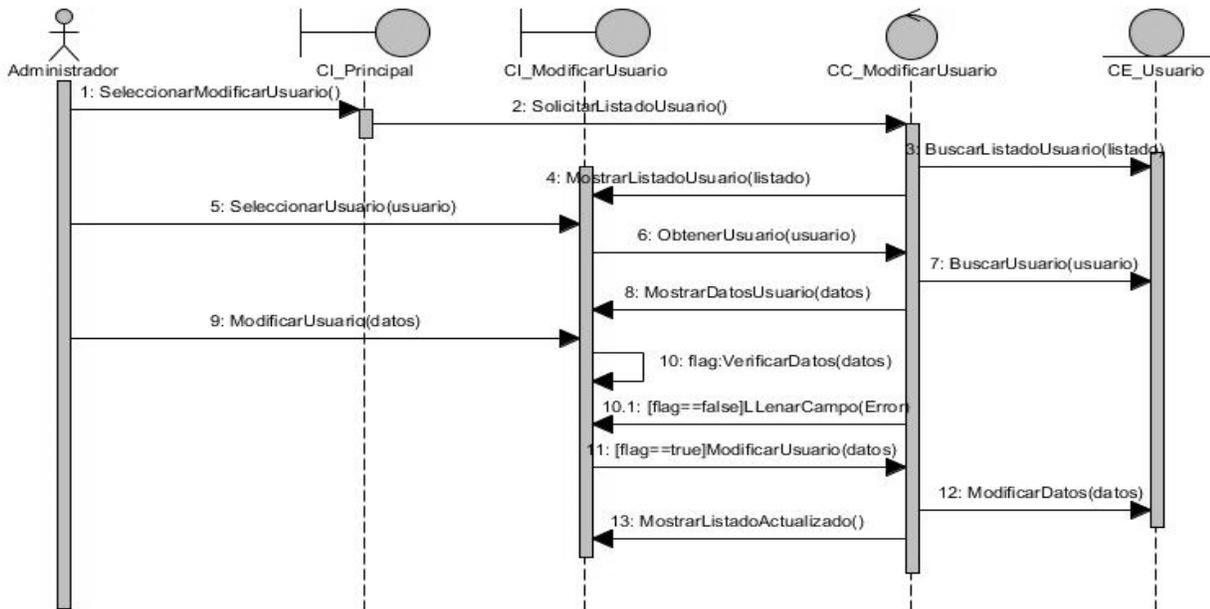


Figura 7 Diagrama de secuencia. RF Modificar Usuario

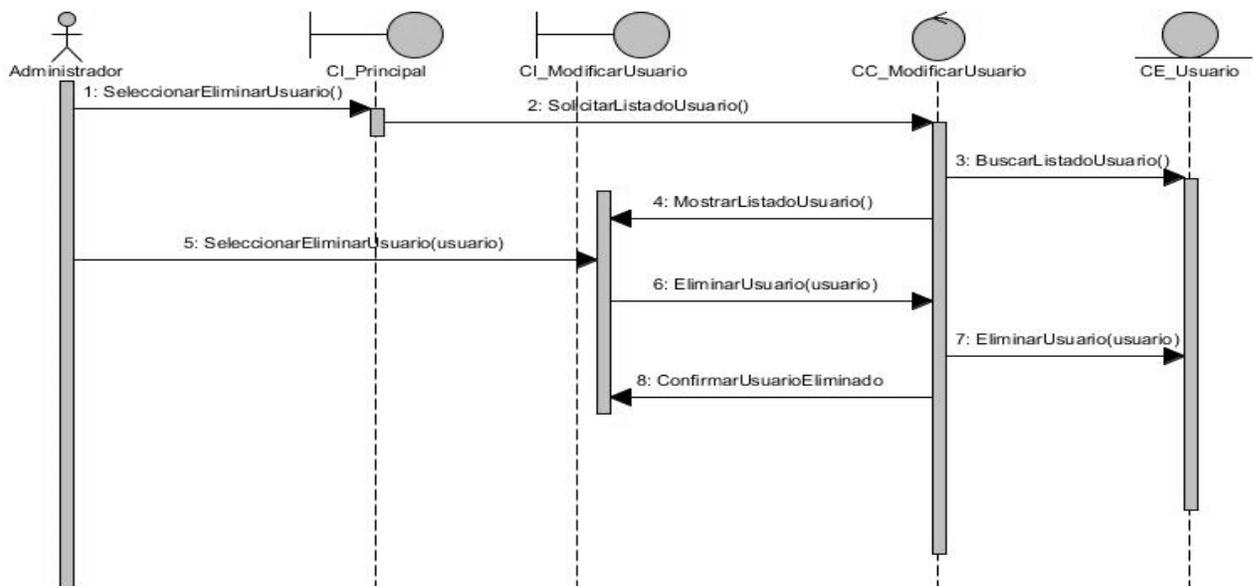


Figura 8 Diagrama de secuencia. RF Eliminar Usuario

2.9 Modelo de Despliegue

El diagrama de despliegue es utilizado para capturar los elementos de configuración del procesamiento y las conexiones entre dichos elementos. Es utilizado también para visualizar la distribución de los componentes de *software* en los nodos físicos. La relación entre los nodos es denominada como protocolos de comunicación.

El portal web necesita para su aplicación satisfactoria un servidor de Base de Datos (BD), un servidor Web donde estarán alojadas las aplicaciones incluyendo el portal web de la UJC y una PC Cliente para que el usuario pueda acceder a los recursos del sitio.

PC Cliente: Es la computadora mediante la cual los usuarios podrán acceder al portal. Esta PC se conecta vía HTTPS con el servidor Web a través del puerto 443.

PC Servidor: Realiza la función de intermediario entre las PC Clientes que realizan sus peticiones y el servidor de BD donde se encuentran las informaciones. Se comunica con el servidor de BD mediante el protocolo TCP por el puerto 5432.

PC Servidor BD: Contiene todas las bases de datos de las funcionalidades del portal web de la UJC.

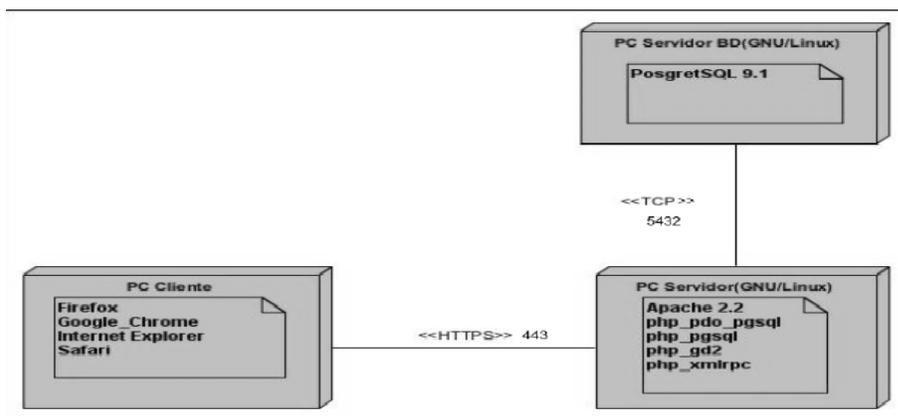


Figura 9 Diagrama de Despliegue

2.10 Conclusiones Parciales

- En el presente capítulo han sido descritas las características del portal. Se expuso la propuesta del sistema que permitió visualizar las principales funcionalidades para una mayor comprensión.
- Se listaron los requerimientos funcionales y no funcionales obtenidos a partir del proceso de identificación de requisitos con el objetivo de satisfacer las necesidades del cliente.
- La modelación de los diferentes diagramas como son: clases del diseño, paquetes, despliegue y casos de uso constituyeron una guía fundamental para la construcción de la propuesta de solución.
- Los patrones de diseño escogidos, propiciaron una arquitectura sólida y robusta.

Capítulo 3. Implementación y Prueba del Sistema

En el presente capítulo se abordarán todos los temas relacionados con las etapas de implementación y prueba del portal de la UJC, quedando definido el diagrama de componente que brinda los detalles principales de la distribución del sistema y los estándares de codificación empleados durante la implementación. Además, se describen los casos de prueba a los que fueron sometidas las funcionalidades del sistema en cada una de las iteraciones.

3.1 Implementación

El modelo de implementación permite organizar los componentes de acuerdo con los mecanismos de estructuración disponible en el entorno de implementación y en el lenguaje utilizado, y muestra la dependencia entre componentes (GIUGNI, y otros, 2010).

3.1.1 Diagrama de componentes

Los diagramas de componentes son utilizados para estructurar el modelo de la implementación. Estos muestran las dependencias lógicas entre componentes de software, y se relacionan con los diagramas de clases ya que un componente normalmente se corresponde con una o más clases, interfaces o colaboraciones. A continuación se representa dicho diagrama: (Alva, 2008)

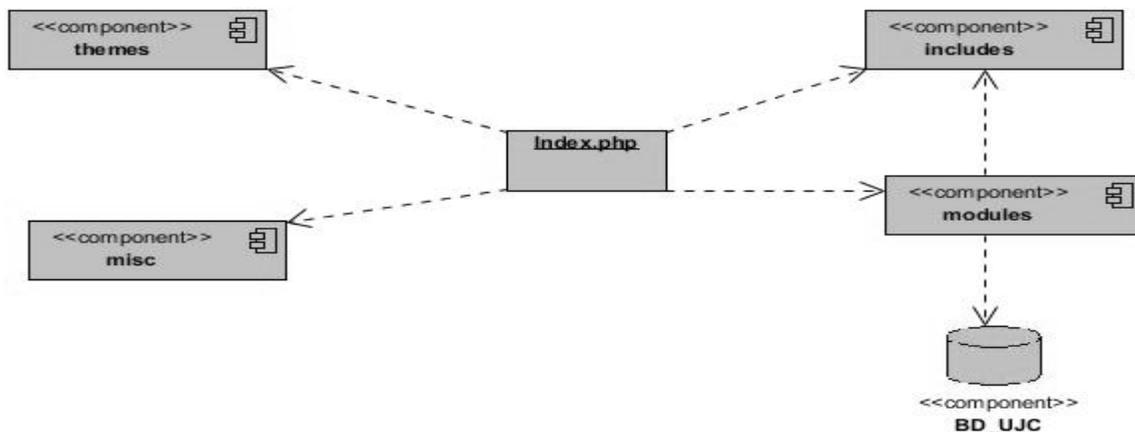


Figura 10 Diagrama de componentes

3.1.2 Descripción de componentes

La tabla que se muestra a continuación describe cada uno de los componentes representados en el diagrama.

Tabla 5 Descripción de componentes

Componente	Descripción
Index.php	Este componente es el punto de inicio de la aplicación, a partir de esta entrada se solicitan los diferentes módulos del CMS Drupal.
Themes	Incluye las distintas plantillas con que trabaja la aplicación, cuando se desee incluir un nuevo diseño, se copia la plantilla en esta carpeta.
Misc	Incluye elementos que tienen que ver con el diseño y funcionamiento (imágenes, ficheros <i>JavaScript</i> , entre otros).
Includes	Este componente tiene un conjunto de ficheros indispensables para el funcionamiento del CMS Drupal.
Modules	Incluye todos los módulos, que permiten las distintas funcionalidades del CMS, cuando desee incorporar un nuevo módulo solo hay que copiarlo en dicha carpeta.
BD_UJC	Representa la Base de Datos de la

	aplicación.
--	-------------

3.2 Estándares de codificación

Los estándares de codificación son un elemento fundamental en la implementación de proyectos, permitiendo que el código generado sea fácil de leer y modificar independientemente de quien ha sido su autor. Son una guía para el equipo de desarrollo, permiten asegurar que el código presente alta calidad y no contenga errores. Drupal le brinda a sus desarrolladores una serie de normas o pautas para fomentar el código de una forma común para todos. A continuación se describen los estándares de codificación que establece Drupal y que fueron utilizados en la implementación de los módulos (Gil, 2012).

Etiquetas de apertura y cierre de PHP

- Cuando estemos escribiendo en PHP, siempre se deben utilizar las etiquetas `<?php` y `?>`, y en ningún caso la versión corta `<? y ?>`.
- En los ficheros `.module` y `.inc` se omite la etiqueta de cierre de PHP (`?>`).
- En los archivos de plantilla `.tpl.php`, cada fragmento de PHP debe llevar sus correspondientes etiquetas de apertura y cierre, para diferenciarlo del código HTML.

Operadores

Los operadores binarios, que se utilizan entre dos valores, deben separarse de estos valores, a ambos lados del operador, por un espacio. Por ejemplo, `$valor = 3`, en lugar de `$valor=3`. Esto se aplica a operadores como `+`, `-`, `*`, `/`, `=`, `==`, `!=`, `>`, `<`, `.` (Concatenación de cadenas), `.=`, `+=`, `-=`, etc.

Los operadores unarios como `++`, `--` no deben tener separación. Por ejemplo, `$valor++`.

Estructuras de control

Con respecto a las estructuras de control, hay que tener en cuenta las siguientes normas:

- Debe haber un espacio entre el comando que define la estructura (*if*, *while*, *for*, etc.) y el paréntesis de apertura. Esto es así para no confundir las estructuras de control con la nomenclatura de las funciones, como veremos más adelante.
- La llave de apertura `{` se situará en la misma línea que la definición de la estructura, separada por un espacio.

- Se recomienda usar siempre las llaves {} aún en los casos en que no sea obligatorio su uso (una sola "línea" de código dentro de la estructura de control).

Funciones

Los nombres de las funciones deben estar escritos en minúsculas y las palabras separadas por guion bajo.

3.3 Pruebas

Las pruebas son los procesos que permiten verificar y revelar la calidad de un producto de software y pueden ser aplicadas periódicamente. Son utilizadas para identificar posibles fallos de implementación, calidad, o usabilidad de un software. Existen varias estrategias de pruebas que suelen ser utilizadas por los equipos de probadores, dentro de las que se pueden mencionar:

Pruebas de Caja Negra

Las Pruebas de Caja Negra se desarrollan sobre la interfaz visual del software y se encarga de verificar que las funciones que debe desempeñar el sistema son operativas. Se centran en los requisitos funcionales de la aplicación, sin internarse en el funcionamiento interno de la misma. Mediante la realización de estas pruebas se pueden encontrar errores de interfaz, funciones incorrectas, errores de salida y problemas con el acceso a datos (Oré, 2009).

Pruebas de Sistema

Tienen como objetivo fundamental verificar el sistema de software para ver si este cumple con sus requisitos. Dentro de esta fase pueden desarrollarse distintos tipos de pruebas, de las cuales algunas son funcionales, prueba de usabilidad, pruebas de rendimiento, pruebas de seguridad, entre otras. El trabajo es centrado fundamentalmente en pruebas funcionales de aplicaciones con interfaces gráficos, las cuales verifican que el software ofrece a los actores humanos la funcionalidad recogida en su especificación. (Febles, 2011)

3.3.1 Funcionales

Para la realización de estas pruebas se utilizó la técnica de caja negra. Estas son realizadas desde el punto de vista de interfaz de usuario, lo cual permite valorar el funcionamiento de la aplicación según la interacción del usuario con el portal (Oré, 2009) (Serna, 2011).

A continuación se muestra el diseño de casos de pruebas correspondiente a la funcionalidad gestionar usuario. El resto de los casos de prueba se pueden encontrar en el Anexo 6.

Condición de ejecución

El usuario accede a la página principal

SC1 <Gestionar Usuario>

Tabla 6 Diseño de casos de prueba Gestionar usuario

Escenario	Descripción	Nombre de usuario	Dirección de correo electrónico	Contraseña	Confirmar contraseña	Roles	Respuesta del sistema	Flujo central
EC 1.1 Insertar usuario	El sistema permite insertar un usuario.	Admin	admin@uci.cu	Admin	Admin	Administrador	Se muestra el mensaje: Se creó una nueva cuenta de usuario para administrador	1-El administrador accede a Usuario que se encuentra en la parte superior del portal. 2-Selecciona la opción añadir usuario 3-Llenar los campos correspondientes y seleccionar la opción crear nueva cuenta.
		Admin	admin@uci.cu		Admin	Administrador	Se muestra el mensaje: * El campo de contraseña es obligatorio.	
			admin@uci.cu	Admin	Admin	Administrador	Se muestra el mensaje: * El	

						dor	campo nombre de usuario es obligatorio.	
		Admin		Admin	Admin	Administrador	Se muestra el mensaje: * El campo dirección de correo electrónico es obligatorio.	
		Admin	admin@uci.cu	Admin		Administrador	Se muestra el mensaje: * El campo de confirmar contraseña es obligatorio.	
EC 1.2	El sistema debe permitir modificar un usuario.	Administrador		Nueva	Nueva		Se muestra el mensaje: Se han guardado los cambios.	<p>1-El administrador accede a la pestaña Usuario.</p> <p>2-Selecciona del listado de usuarios existentes el usuario que desea modificar.</p> <p>3-Selecciona la opción editar.</p> <p>4-Llena los campos correspondientes y selecciona la opción guardar.</p>
EC 1.3	El sistema	N/A		N/A			El sistema	1-El administrador

Eliminar usuario	debe permitir eliminar un usuario					muestra el mensaje “(nombre de la cuenta)” se ha desactivado	accede a Usuarios. 2-Selecciona el usuario que desea eliminar. 3-Selecciona la opción editar. 4-Selecciona la opción de cancelar cuenta. 5-Marcar los parámetros que desea mantener o eliminar y seleccionar la opción cancelar cuenta.
EC 1.4 Mostrar usuario	El sistema debe permitir mostrar los datos de un usuario.	N/A		N/A		Se muestran todos los usuarios existentes en el sistema	1-Acceder a Usuarios.

Resultados de las pruebas de funcionalidad

Se probaron todos los casos de pruebas que responden a las funcionalidades del portal web de la UJC. En la total fueron detectadas 25 no conformidades de las cuales todas fueron resueltas, donde los principales errores detectados se deben a errores ortográficos y textos en idiomas diferentes del español. A continuación se muestra una gráfica donde se desglosa las no conformidades detectadas en las dos iteraciones realizadas. En la primera iteración se recoge un total de 20 no conformidades y una segunda iteración arrojó un total de 5 no conformidades quedando todas resueltas.

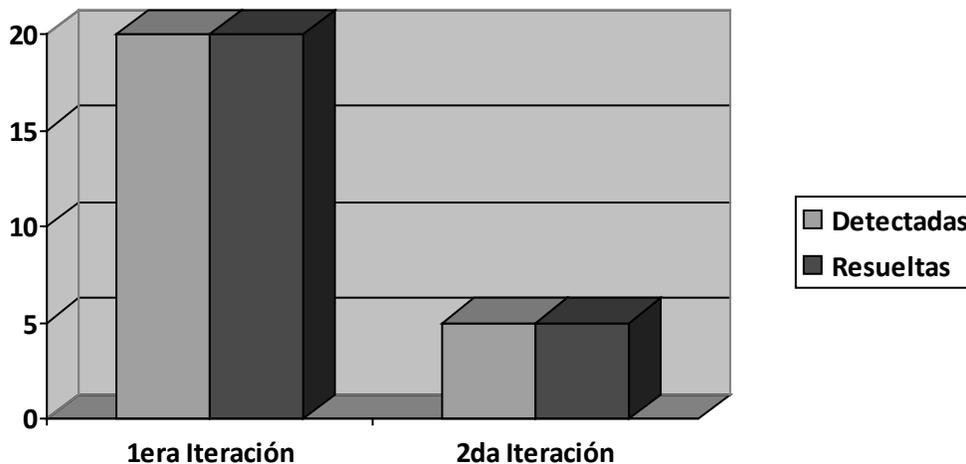


Figura 11 Resultado de las pruebas funcionales

3.3.2 Seguridad

Las pruebas de seguridad permiten realizar una evaluación de los sistemas desde el punto de vista externo y sin conocimiento previo del mismo. Tienen como objetivo hacer un análisis con el fin de encontrar fallos de seguridad tanto en el diseño como en la implementación de la aplicación. Además buscan medir la confidencialidad, integridad y disponibilidad de los datos, partiendo de la identificación de amenazas y riesgos en el uso de interfaces de usuarios final. Una vez terminadas las pruebas es posible medir y cuantificar los riesgos a los cuales se ven expuestos aplicativos de la infraestructura interna y externa (vyvquality, 2012) .

Resultados de las pruebas de seguridad

Las pruebas de seguridad se aplicaron hasta 2 niveles. En el nivel 1 se confeccionó el documento “Listas de Chequeo de Pruebas de Seguridad”, midiendo 4 indicadores. Ver Anexo 7.

En el nivel 2 se desarrollaron las pruebas con ayuda de la herramienta *Acunetix Web Vulnerability Scanner 8.0 0*, que establece alertas de tipo: alta, media, baja e informativa.

El nivel 1 en la primera iteración arrojó un total de 3 elementos afectados para 3 no conformidades críticas quedando evaluada de insatisfactoria la prueba. Las no conformidades encontradas fueron solucionadas

en su totalidad. En la segunda iteración la prueba fue evaluada de satisfactoria, debido a que no se encontraron no conformidades.

En la primera iteración del nivel 2 se detectaron a través de la herramienta empleada 74 no conformidades medias, 2 no conformidades bajas y 40 no conformidades informacionales para un total de 106 no conformidades. En la segunda iteración no se encontraron no conformidades que afecten la seguridad de la aplicación. A continuación se muestra un informe con los tipos de vulnerabilidades detectadas por la aplicación y su posible solución:

Tabla 7 Informe de la herramienta WebSecurity

Vulnerabilidad	Solución
<p>Varias rutas del sistema se dieron a conocer dentro del código fuente de la aplicación del cliente a de otros archivos. Esta información podría ser utilizada por atacantes para hacer una conjetura sobre el entorno de la aplicación y las debilidades heredadas que pueden venir con ellas</p>	<p>Configurar el apache (directorio alias) Borrar el archivo info.php</p>
<p>Divulgación banner</p>	<p>En Apache: configurar el fichero: "/etc/apache2/apache2.conf", cambiar de ServerSignature on a ServerSignature Off y también cambiar de ServerTokens full a ServerTokens Prod. O configurar el fichero: "/etc/apache2/conf.d/security" con los mismos parámetros.</p>

3.3.3 Carga

Usada para validar y valorar la aceptabilidad de los límites operacionales de un sistema bajo carga de trabajo variable, mientras el sistema bajo prueba permanece constante; se simula la carga de trabajo promedio y con picos que ocurre dentro de tolerancias operacionales normales (Fraile, 2011).

Resultados de las pruebas de carga

Para la realización de las pruebas de carga se efectuó un análisis del tráfico de usuarios en los sistemas homólogos estudiados, lo que arrojó que las visitas a los portales web se encontraban en un rango de 3000 a 5000 visitas mensuales, lo que sería aproximadamente 100 usuarios diarios.

Las pruebas de carga se desarrollaron con la ayuda de la herramienta *Apache JMeter*, en un ambiente utilizando un ordenador con las siguientes características:

- Sistema operativo Ubuntu 13.04.
- Microprocesador Intel Core 2Duo a 2.20GHz.
- Memoria RAM 3GB.

Los resultados obtenidos en las pruebas de carga se consideran satisfactorios debido a que los tiempos de respuestas del servidor se encuentran en el rango de tiempo (de 1 a 5 segundos) que se especificó en los requerimientos del sistema. Además se probó para un total de 50 y 100 usuarios concurrentes, obteniendo los resultados que se muestran en la Figura 13.

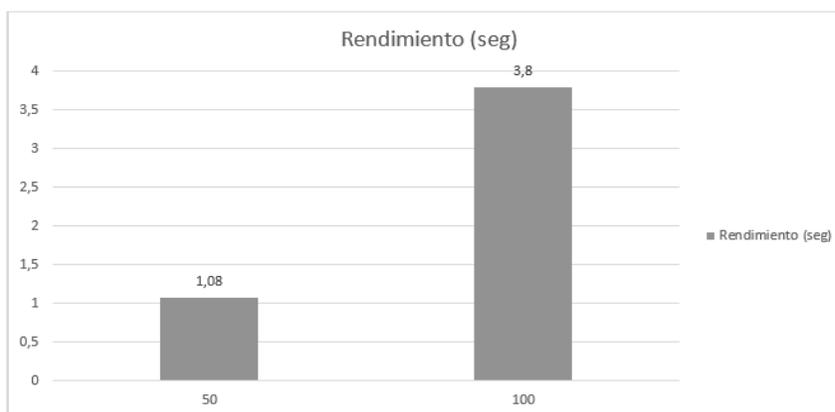


Figura 12 Resultados de las pruebas de carga

3.3.4 Estrés

Enfocada a evaluar cómo el sistema responde bajo condiciones anormales. (Extrema sobrecarga, insuficiente memoria, servicios y hardware no disponible, recursos compartidos no disponible).

Resultados de las pruebas de estrés

Las pruebas de estrés se desarrollaron con la ayuda de la herramienta Apache *JMeter*. Los resultados de las pruebas de estrés se consideran satisfactorios, pues luego de sobrepasar la cantidad de 100 usuarios con un total de 150 y 200 usuarios concurrentes, la propuesta de solución se mantuvo estable prestando servicios todo el tiempo sin incurrir en fallos, arrojando los resultados que se muestran en la Figura14.

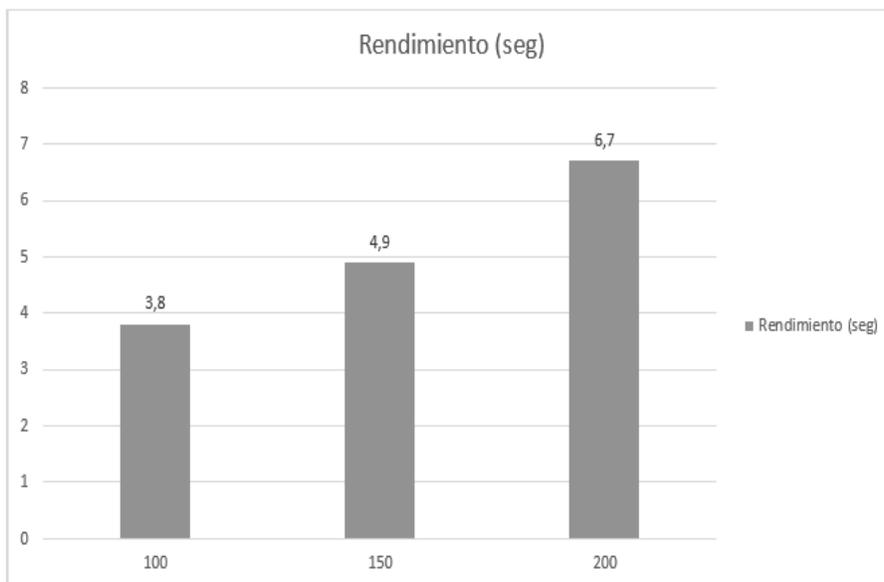


Figura 13 Resultados de las pruebas de estrés

3.3.5 Usabilidad

Evaluar la usabilidad de un sitio, permiten identificar los elementos y sus niveles de compromisos con el cumplimiento de los requisitos de efectividad, eficiencia y satisfacción para un contexto de uso determinado (Florian, 2010).

Resultados de las pruebas de usabilidad

Las pruebas de usabilidad aplicadas a la propuesta de solución se realizaron aplicando el documento “Lista de Chequeo de Usabilidad de Sitios Web”. Esta cuenta con 110 indicadores separados por 13 elementos de usabilidad, la cual se puede observar en el Anexo 8.

Se efectuaron en dos iteraciones. En la primera iteración quedó evaluada de regular la prueba, debido a que se encontraron 7 no conformidades críticas y 1 no conformidades no críticas, para un total de 8 no conformidades y con 8 elementos afectados. En la segunda iteración fueron solucionadas todas las no conformidades, resultando satisfactoria la prueba.

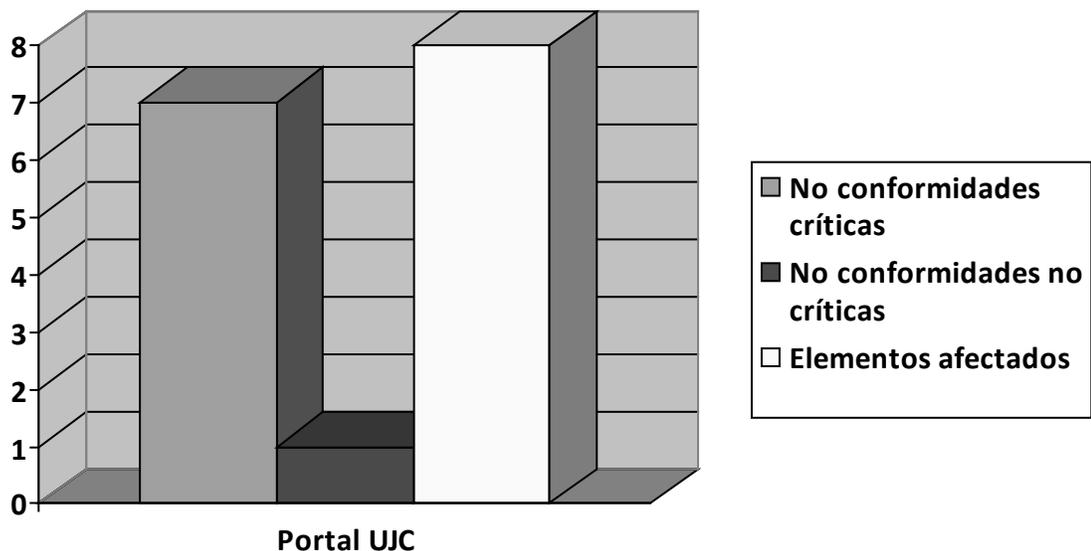


Figura 14 Resultados pruebas de usabilidad

3.4 Pantallas de la aplicación

Las pantallas principales del portal web de la Unión de Jóvenes Comunistas (UJC) son el resultado visual donde se define la apariencia de cada una de las páginas del sitio, ubicando en ellas todos los elementos que las componen. A continuación se muestra la pantalla principal del portal web de la UJC:

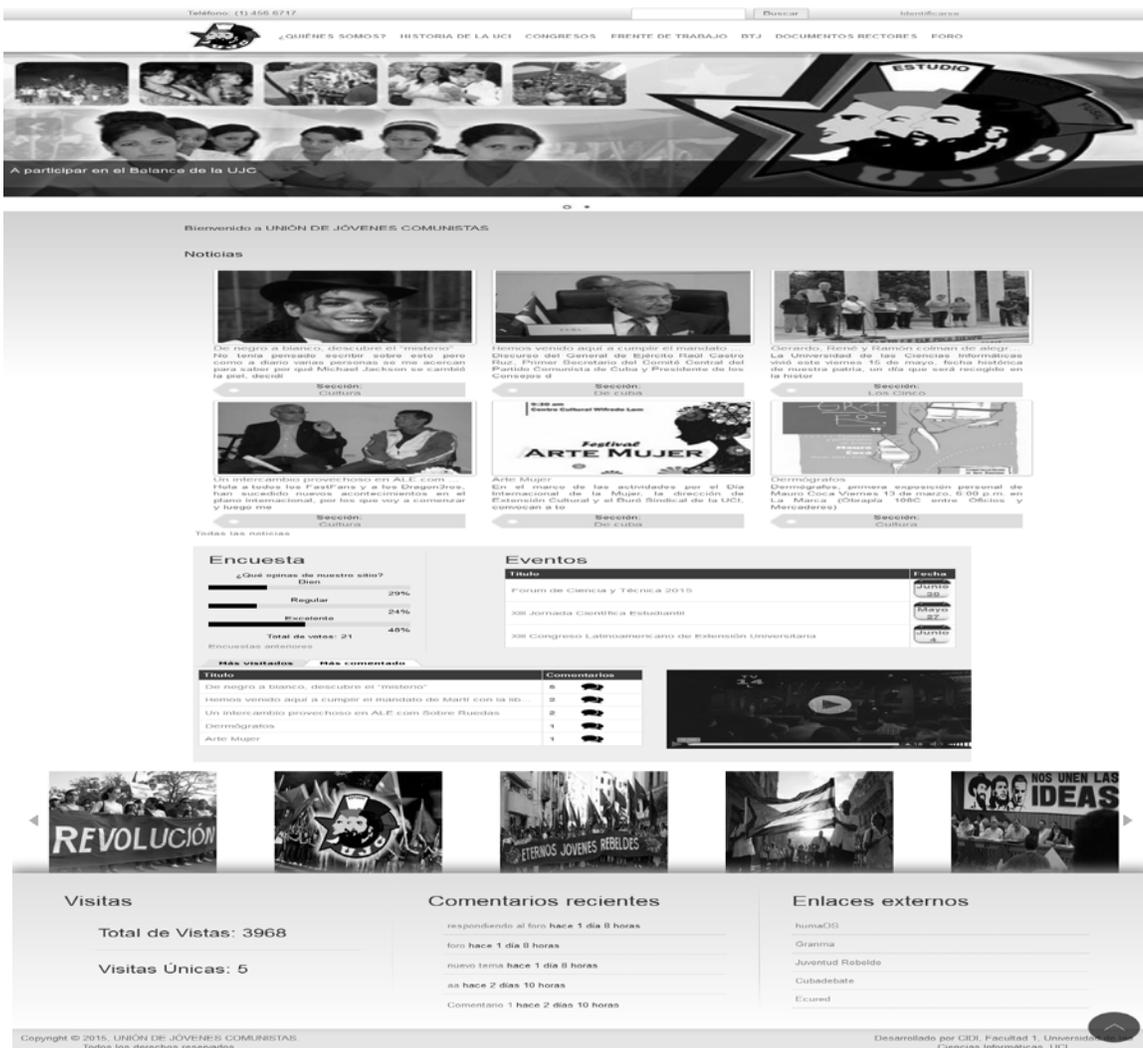


Figura 15 Pantalla de la aplicación

3.5 Conclusiones del capítulo

- En este capítulo la confección del diagrama de componentes ofreció una vista arquitectónica de alto nivel para ayudar al equipo de desarrollo en la implementación.
- Al mismo tiempo la definición de los estándares de codificación a tener en cuenta para la implementación del portal web, permitió garantizar que el código posea alta calidad, menos errores y que pueda ser mantenido fácilmente; así como reutilizado por desarrolladores de otros proyectos que lo necesiten.
- Se mostraron las principales pantallas de la aplicación lo cual visualiza como quedó implementado el sistema.
- La realización de las pruebas de software al portal web permitió encontrar los errores existentes para poder solucionarlos, posibilitando determinar y asegurar la calidad y robustez del portal web de la UJC.

Conclusiones

El desarrollo del presente trabajo posibilitó el cumplimiento de los objetivos y tareas propuestas, por lo que se arriban a las siguientes conclusiones:

- Se identificaron las funcionalidades que debía cumplir el sistema a partir de un estudio de soluciones similares lo que permitió identificar los elementos que podían ser utilizados para la implementación de la solución. Además posibilitó seleccionar la metodología, herramientas y tecnologías que serían utilizadas para cumplir con las necesidades de gestión de la información por parte de los clientes.
- La captura de requisitos y la modelación de los diferentes diagramas permitió un mejor análisis y diseño de las funcionalidades desarrolladas para el portal de la UJC constituyendo una guía fundamental para la construcción de la propuesta de solución.
- Como resultado de la implementación y dándole cumplimiento al objetivo general de la investigación se obtuvo un portal web para la UJC de la UCI, que permitirá la divulgación y promoción del trabajo de esta organización en la universidad.
- La realización de las pruebas de software permitieron encontrar los errores existentes para poder solucionarlos, posibilitando determinar y asegurar la calidad y robustez del portal web de la UJC.

Recomendaciones

Para futuras versiones del portal web se recomienda:

- Continuar agregándole funcionalidades al portal web según las necesidades que se requieran para mejorar el proceso de gestión de la información, teniendo en cuenta los avances en las TIC.
- Capacitación a los usuarios que van a administrar la aplicación para un mejor entendimiento de las funcionalidades con las que cuenta el sistema.
- Brindar las posibilidades de navegar en el portal en distintos idiomas para lograr la internacionalización de las publicaciones.
- Optimizar el diseño del portal web para dispositivos móviles.
- Presentar el trabajo en futuros eventos científicos e investigativos y trabajar en una posible publicación para revista referenciada.

Referencias

AJA QUIROGA, Lourdes Acimed. 2002. *Gestión de información, gestión del conocimiento y gestión de la calidad en las organizaciones.* 2002. Vol. 10.

ALMENARA, Julio Cabero. 2005. *Las TICs y las Universidades: retos, posibilidades y preocupaciones.* Rev. Educ. Super : s.n., 2005. Vol. 34.

Alva, Rivera. 2008. Arquitectura de Software II. Diagramas de Componentes y Despliegue. . [En línea] 11 de Noviembre de 2008. <http://es.scribd.com/doc/7884665/Arquitectura-de-Software-II-Diagrama-de-Componentes-y-Despliegue>.

Amaro, S D y Valverde, J C. 2007. *Metodologías Ágiles.* s.l. : Universidad Nacional de Trujillo. Tesis de Pregrado, 2007.

BERNOT, Y M. 2012. Sistema Gestor de Base de Datos. [En línea] Noviembre de 2012. http://www.ecured.cu/index.php/Sistema_Gestor_de_Base_de_Datos.

Buschmann. 1996. *Pattern - Oriented Software Architecture. A System of Patterns.* 1996.

Caballero Redondo, María Jose. 2012. Análisis de aplicación: JMeter. [En línea] Marzo de 2012. [Citado el: 22 de febrero de 2015.] http://www.bilib.es/fileadmin/user_upload/analisis-bilibjvmeter-v0.2.pdf. .

Clements, Paul. Software Architecture in Practice. [En línea] [Citado el: 05 de febrero de 2015.] <http://books.google.com.cu/books?hl=es&lr=&id=mdilu8Kk1WMC&oi=fnd&pg=PR11&dq=Clements,+Paul.+Software+Architecture+in+Practice&ots=UcP0N7jc>.

Cuba, Soy. 2013. Soy Cuba. [En línea] 2013. [Citado el: 10 de mayo de 2015.] <http://soycuba.cu/>.

DE LA LENGUA, Real Academia. 1992. Diccionario de la Lengua Española, vol. I. Real Academia Española. [En línea] 1992.

Douglas, Korry. 2005. *PostgreSQL.* USA : Sams Indianapolis, 2005. 0672327562.

Dragones. 2010. Dragones. [En línea] 2010. [Citado el: 10 de mayo de 2015.] <https://dragones.uci.cu/>.

ECUADOR, PARTIDO COMUNISTA DEL. 2012. PARTIDO COMUNISTA DEL ECUADOR. [En línea] 2012. <http://www.pcecuador.org/page/index.php>.

ESPINOZA LEÓN, Claudia Marina. 2010. Método de Integración y desarrollo de software hacia una arquitectura orientada a servicios para el Departamento Informático de la Universidad de Cuenca. [En línea] 2010. [Citado el: 15 de febrero de 2015.] dspace.ucuenca.edu.ec/handle/123456789/2526.

Febles Estrada, Ailyn , y otros. 2011. *Una experiencia novedosa para el testing desarrollada por un departamento de pruebas de software.* *Revista Cubana de Ciencias Informáticas.* 2011. Vol. Vol 5 N° 2. .

Florian, Beatriz E., Solarte, Osbaldo y Reyes , Javier M. 2010. Propuesta para incorporar evaluación y pruebas de usabilidad dentro de un proceso de desarrollo de software. [En línea] Julio de 2010. http://www.scielo.org.co/scielo.php?pid=S1794-12372010000100010&script=sci_arttext..

Foundation, The Apache Software. 2011. The Apache HTTP Server Project. [En línea] 20 de 05 de 2011. [Citado el: 11 de noviembre de 2014.] [http://httpd.apache.org/..](http://httpd.apache.org/)

Fraile, Luis. 2011. Pruebas de rendimiento. [En línea] 2011. [Citado el: 12 de abril de 2015.] [http://www.globetesting.com/pruebas-de-rendimiento/..](http://www.globetesting.com/pruebas-de-rendimiento/)

Foundation, Eclipse. 2012. OpenUP. [En línea] 30 de 05 de 2012. [Citado el: 02 de diciembre de 2014.] [http://epf.eclipse.org/wikis/openup/..](http://epf.eclipse.org/wikis/openup/)

Gamma, Erich. 2005. *Patrones de Diseño.* 2005.

Gil, Fran. 2012. Experto en Drupal 7 Nivel avanzado. Curso de creación y gestión de portales web con Drupal 7. . [En línea] 17 de octubre de 2012.

GIUGNI, Marylin y LOAIZA, Reina. 2010. *Metodología para el desarrollo de portales centrada en el usuario: una evaluación empírica.* 2010. Vol. 7.

Gómez, A C. 2011. XHTML. [En línea] 2011. <http://www.ecured.cu/index.php/XHTML>.

HECHEVARRIA, R G. 2011. CSS. [En línea] 2011. <http://www.ecured.cu/index.php/CSS>.

Henst, C V D. 2011. ¿Qué es el PHP? [En línea] enero de 2011. [Citado el: 20 de noviembre de 2014.] [http://www.maestrosdelweb.com/editorial/phpintro/.](http://www.maestrosdelweb.com/editorial/phpintro/)

Heredia Ruiz, Javier, Álvarez Almanza, Lilian y Linares Pons, Naryana. 2011. *Comparación y tendencias entre metodologías ágiles y formales*. La Habana : s.n., 2011. Serie Científica de la Universidad de las Ciencias Informáticas.

Jaliet Bárbara Rojas Hernández, Daylin Matos Castillo. 2008. *Sitio de la UJC en la UCI*. s.l. : Trabajo de Diploma de Pregrado de la UCI, 2008.

LARMAN, Craig. 2002. UML y Patrones. s.l. : Segunda Edición. s.l., 2002.

Ltd, Enterprise Architect Sparx Systems Pty. 2011. Herramienta CASE para diseño con UML y desarrollo de software. [En línea] 19 de 09 de 2011. [Citado el: 11 de noviembre de 2014.] <http://www.sparxsystems.com.ar/products/ea/index.html>.

Martínez, Ander. 2009. Apache JMeter. Manual de usuario v1.2. [En línea] 20 de 05 de 2009. [Citado el: 8 de abril de 2015.] <http://www.ejie.net/documentos/Herramientas/JMeter.%20Manual%20de%20usuario%20v1.2.pdf> .

Muras, Miguel Angel. 2009. Moisaic tecnología y comunicación multimedia. [En línea] 23 de diciembre de 2009. [Citado el: 15 de mayo de 2015.] <http://mosaic.uoc.edu/2009/12/23/una-aproximacion-a-los-gestores-de-contenidos-cms/>..

Oliveros, Alejandro y del Valle Rojo, Silvana. 2012. Requerimientos No funcionales para aplicaciones Web. [En línea] 2012. [Citado el: 5 de abril de 2015.] http://www.41jaiio.org.ar/sites/default/files/432_ASSE_2012.pdf ISSN: 1850-2792.

Oracle. 2014. About Java. [En línea] 2014.

Oré B, Ing. Alexander. 2009. Pruebas funcionales. [En línea] 2009. [Citado el: 23 de marzo de 2015.] http://www.calidadyssoftware.com/testing/pruebas_funcionales.php..

Paraguay, Partido Comunista. 2013. Partido Comunista Paraguayo. [En línea] 2013. [Citado el: 10 de mayo de 2015.] <http://pcparaguay.org/v2/>.

Piattini, M. 1996. *Análisis y Diseño Detallado de Aplicaciones Informáticas de Gestión*. Madrid : Rama, 1996.

PISCITELLI, Alejandro. 2005. *Internet, la imprenta del siglo XXI.* 2005.

Rebelde, Diario de la Juventud. 2002. Diario de la Juventud Rebelde. [En línea] 2002. [Citado el: 10 de mayo de 2015.] <http://www.juventudrebelde.cu/>.

SÁNCHEZ, I.B. 2014. *Aplicación de soporte a la metodología de desarrollo de distribuciones GNU/ LINUX, Nova-OpenUp.* Universidad de las Ciencias Informáticas, La Habana : Trabajo de diploma para optar por el título de ingeniero en Ciencias Informáticas, 2014.

Santamaría, Israel Fausto Valencia. 2011. *COMPARATIVA DE GESTIONADORES DE CONTENIDO (CMS) PARA LA CONSTRUCCIÓN DE SITIOS WEB.* Ibarra : Universidad Técnica del Norte, 2011.

Serna M., Edgar. 2011. Análisis y comparación de las propuestas recientes para diseñar casos de pruebas desde casos de usos orientados a verificar los aspectos del funcionamiento del software. [En línea] 2011. http://www.bdigital.unal.edu.co/6114/1/71617999._2012.pdf. .

SHREVES, R. 2008. Open Source CMS Market Share. [En línea] 2008. [Citado el: 25 de octubre de 2014.] <http://www.waterandstone.com/downloads/2008OpenSourceCMSMarketSurvey.pdf>.

UJC, BURÓ NACIONAL DE LA. 2014. *REGLAMENTO PARA LAS ORGANIZACIONES DE BASE.* La Habana : UNION DE JOVENES COMUNISTAS COMITÉ NACIONAL, 2014.

Unión de Juventudes Comunistas de España. 2012. Unión de Juventudes Comunistas de España. [En línea] 2012. [Citado el: 10 de mayo de 2015.] <http://www.agitacion.org/>.

Venezuela, Juventud Comunista de. 2013. Juventud Comunista de Venezuela. [En línea] 2013. [Citado el: 10 de mayo de 2015.] <https://prensapcv.com>.

vyvquality.com. 2012. Pruebas de seguridad. [En línea] 2012. [Citado el: 25 de marzo de 2014.] <http://www.vyvquality.com/w1/index.php/servicios/pruebas-de-seguridad.html>..

Bibliografía

AJA QUIROGA, Lourdes Acimed. 2002. *Gestión de información, gestión del conocimiento y gestión de la calidad en las organizaciones.* 2002. Vol. 10.

ALMENARA, Julio Cabero. 2005. *Las TICs y las Universidades: retos, posibilidades y preocupaciones.* Rev. Educ. Super : s.n., 2005. Vol. 34.

Alva, Rivera. 2008. Arquitectura de Software II. Diagramas de Componentes y Despliegue. . [En línea] 11 de Noviembre de 2008. <http://es.scribd.com/doc/7884665/Arquitectura-de-Software-II-Diagrama-de-Componentes-y-Despliegue>.

Amaro, S D y Valverde, J C. 2007. *Metodologías Ágiles.* s.l. : Universidad Nacional de Trujillo. Tesis de Pregrado, 2007.

BERNOT, Y M. 2012. Sistema Gestor de Base de Datos. [En línea] Noviembre de 2012. http://www.ecured.cu/index.php/Sistema_Gestor_de_Base_de_Datos.

Burbeck, Steve. Applications Programming in Smalltalk-80(TM): How to use Model-View- Controller (MVC). [En línea] Noviembre 21, 2014. Disponible en: <http://st-www.cs.illinois.edu/users/smarch/st-docs/mvc.html>

Buschmann. 1996. *Pattern - Oriented Software Architecture. A System of Patterns.* 1996.

Caballero Redondo, María Jose. 2012. Análisis de aplicación: JMeter. [En línea] Marzo de 2012. [Citado el: 22 de febrero de 2015.] http://www.bilib.es/fileadmin/user_upload/analisis-bilibjvmeter-v0.2.pdf.

Chávez, Michel AriasRevista Intersedes. 2006. La ingeniería de requerimientos y su importancia en el desarrollo de proyectos de software. San José: Universidad de Costa Rica, 2006

Clements, Paul. Software Architecture in Practice. [En línea] [Citado el: 05 de febrero de 2015.] <http://books.google.com.cu/books?hl=es&lr=&id=mdilu8Kk1WMC&oi=fnd&pg=PR11&dq=Clements,+Paul.+Software+Architecture+in+Practice&ots=UcP0N7jc>.

Cuba, Soy. 2013. Soy Cuba. [En línea] 2013. [Citado el: 10 de mayo de 2015.] <http://soycuba.cu/>.

- DE LA LENGUA, Real Academia. 1992.** Diccionario de la Lengua Española, vol. I. Real Academia Española. [En línea] 1992.
- Douglas, Korry. 2005.** *PostgreSQL*. USA : Sams Indianapolis, 2005. 0672327562.
- Dragones. 2010.** Dragones. [En línea] 2010. [Citado el: 10 de mayo de 2015.] <https://dragones.uci.cu/>.
- ECUADOR, PARTIDO COMUNISTA DEL. 2012.** PARTIDO COMUNISTA DEL ECUADOR. [En línea] 2012. <http://www.pcecuador.org/page/index.php>.
- ESPINOZA LEÓN, Claudia Marina. 2010.** Método de Integración y desarrollo de software hacia una arquitectura orientada a servicios para el Departamento Informático de la Universidad de Cuenca. [En línea] 2010. [Citado el: 15 de febrero de 2015.] dspace.ucuenca.edu.ec/handle/123456789/2526.
- Febles Estrada, Ailyn , y otros. 2011.** *Una experiencia novedosa para el testing desarrollada por un departamento de pruebas de software. Revista Cubana de Ciencias Informáticas*. 2011. Vol. Vol 5 N° 2. .
- Flanagan, David.** *JavaScript: The Definitive Guide*. s.l.: O'Reilly, 2006. 0596101996.
- Florian, Beatriz E., Solarte, Osbaldo y Reyes , Javier M. 2010.** Propuesta para incorporar evaluación y pruebas de usabilidad dentro de un proceso de desarrollo de software. [En línea] Julio de 2010. http://www.scielo.org.co/scielo.php?pid=S1794-12372010000100010&script=sci_arttext..
- Foundation, The Apache Software. 2011.** The Apache HTTP Server Project. [En línea] 20 de 05 de 2011. [Citado el: 11 de noviembre de 2014.] <http://httpd.apache.org/>..
- Fraile, Luis. 2011.** Pruebas de rendimiento. [En línea] 2011. [Citado el: 12 de abril de 2015.] <http://www.globetesting.com/pruebas-de-rendimiento/>. .
- Foundation, Eclipse. 2012.** OpenUP. [En línea] 30 de 05 de 2012. [Citado el: 02 de diciembre de 2014.] <http://epf.eclipse.org/wikis/openup/>..
- Gamma, Erich. 2005.** *Patrones de Diseño*. 2005.
- Gil, Fran. 2012.** Experto en Drupal 7 Nivel avanzado. Curso de creación y gestión de portales web con Drupal 7. . [En línea] 17 de octubre de 2012.

GIUGNI, Marylin y LOAIZA, Reina. 2010. *Metodología para el desarrollo de portales centrada en el usuario: una evaluación empírica.* 2010. Vol. 7.

Gómez, A C. 2011. XHTML. [En línea] 2011. <http://www.ecured.cu/index.php/XHTML>.

HECHEVARRIA, R G. 2011. CSS. [En línea] 2011. <http://www.ecured.cu/index.php/CSS>.

Henst, C V D. 2011. ¿Qué es el PHP? [En línea] enero de 2011. [Citado el: 20 de noviembre de 2014.] <http://www.maestrosdelweb.com/editorial/phpintro/>.

Heredia Ruiz, Javier, Álvarez Almanza, Lilian y Linares Pons, Naryana. 2011. *Comparación y tendencias entre metodologías ágiles y formales.* La Habana : s.n., 2011. Serie Científica de la Universidad de las Ciencias Informáticas.

Jaliet Bárbara Rojas Hernández, Daylin Matos Castillo. 2008. *Sitio de la UJC en la UCI.* s.l. : Trabajo de Diploma de Pregrado de la UCI, 2008.

LARMAN, Craig. 2002. UML y Patrones. s.l. : Segunda Edición. s.l., 2002.

LEBRÚN, Carlos Arturo Vega. 2005. Integración de herramientas de tecnologías de información “portales colaborativos de trabajo” como soporte en la administración del conocimiento. Doctorado en Tecnologías de Información y Análisis de Decisiones. Puebla, M: s.n.2005

UJC, BURÓ NACIONAL DE LA. 2014. *REGLAMENTO PARA LAS ORGANIZACIONES DE BASE.* La Habana : UNION DE JOVENES COMUNISTAS COMITÉ NACIONAL, 2014.

Unión de Juventudes Comunistas de España. 2012. Unión de Juventudes Comunistas de España. [En línea] 2012. [Citado el: 10 de mayo de 2015.] <http://www.agitacion.org/>.

Venezuela, Juventud Comunista de. 2013. Juventud Comunista de Venezuela. [En línea] 2013. [Citado el: 10 de mayo de 2015.] <https://prensapcv.com>.

vyvquality.com. 2012. Pruebas de seguridad. [En línea] 2012. [Citado el: 25 de marzo de 2014.] <http://www.vyvquality.com/w1/index.php/servicios/pruebas-de-seguridad.html..>

Glosario de Términos

Apache: Servidor web de distribución libre y de código abierto.

HTTPS: Protocolo de transferencia de hipertexto, es el protocolo usado en cada transacción de la web.

Internet: Internet es la red de redes donde reside toda la información. Es un conjunto de redes, redes de ordenadores y equipos físicamente unidos mediante cables que conectan puntos de todo el mundo.

Patrón de arquitectura: Es una solución a un problema, que se puede aplicar muchas veces, en distintas situaciones.

Rol: Conjunto de acciones que realiza una persona como actor.

Servicio: Acción de una organización destinada a satisfacer necesidades.

Software: Es un término que viene a decir “Parte Lógica” y es el conjunto de elementos lógicos que componen un sistema informático.

URL: Acrónimo de *Uniform Resource Locator*. Localizador de Recursos Uniforme. Es la dirección de una página web.

Web: Sistema lógico de acceso y búsqueda de la información disponible en Internet, cuyas unidades informativas son las páginas web, páginas a las que se puede acceder usando un navegador.