

Universidad de las Ciencias Informáticas

Facultad 1

Título: Portal web del Partido Comunista de Cuba

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

Autor: Leonardo Jesús Gómez Romay

Tutores: Ing. Ezequiel Martínez Justiz

Ing. Adrián Antonio Garbey Fonseca

La Habana, Junio de 2015

“Año 57 de la Revolución”



“Los seres humanos sin conocimientos, se asemejan a un meteoro viajando sin rumbo en el espacio sideral.”

Sidbarto

Dedicatoria

Dedico el presente trabajo de diploma a mis padres y a mis abuelos por su enorme apoyo y sacrificio durante todos estos años, por la educación que me han dado, y por los valores y principios que me han inculcado en toda la etapa de mi vida. A ellos les debo todo lo que soy hoy.

A mi tío Fernando que siempre me ha querido como si fuera su hijo.

A mi hermanita linda que tanto quiero.

A mi amigo Lander que es como un hermano para mí.

A mi esposa Sachie que siempre estuvo presente para darme su ayuda incondicional cuando más lo necesité.

Agradecimientos

Agradezco a mis padres por todo su apoyo a lo largo de estos cinco años, por creer en mí y darme la fuerza necesaria para llegar a ser un profesional.

A mis abuelos que creyeron en mí con orgullo. Pensar en ellos y en la ilusión que siempre tuvieron de verme formado como profesional, me dio más fuerzas para seguir adelante y no defraudarlos.

A mis tutores Adrián y Ezequiel gracias por sus consejos, conocimientos y el tiempo que emplearon en ayudarme.

A mis amistades del aula con las que he compartido buenos y malos momentos a lo largo de estos 5 años, en especial a Víctor y a Castillo.

A toda mi familia en general y a todo el que de una forma u otra me ha ayudado y apoyado en este paso tan importante de mi vida.

A todos gracias.

Declaración de Autoría

Declaro que soy el único autor de este trabajo y autorizo a la Facultad 1 de la Universidad de las Ciencias Informáticas a hacer uso de este en su beneficio.

Para que así conste firmamos la presente a los _____ días del mes de _____ del _____.

Leonardo Jesús Gómez Romay

Firma del Autor

Ing. Adrián Antonio Garbey Fonseca

Firma del Tutor

Ing. Ezequiel Martínez Justiz

Firma del Tutor

El Partido Comunista de Cuba (PCC) constituye la organización política que se alza como la vanguardia organizada de la nación cubana, siendo la fuerza dirigente superior de la sociedad y el Estado. El PCC tiene como una de sus prioridades la constante comunicación con la población cubana y la retroalimentación necesaria que permita el desarrollo de la democracia socialista en el país. Uno de los mecanismos creados para este fin es la publicación de un portal web, donde se encuentran informaciones y datos fundamentales del accionar diario de la organización; pero surge la dificultad de que dicho portal se encuentra desarrollado bajo los principios de la web estática, y su arquitectura y funcionamiento no están acordes al desarrollo tecnológico actual, lo cual dificulta la gestión de la información en este. La presente investigación tiene como objetivo desarrollar un nuevo portal web para el PCC, que facilite el proceso de gestión de la información. El desarrollo del portal web está guiado por las especificaciones que propone la metodología de desarrollo de *software* AUP, que facilitó el análisis, el diseño, la implementación y la validación de las funcionalidades del sistema. La utilización del Sistema de Gestión de Contenidos *Drupal*, como gestor de base de datos PostgreSQL, las herramientas: *Visual Paradigm*, *NetBeans* y *Apache JMeter*, y los lenguajes: *UML*, *JavaScript*, *CSS*, *HTML* y *PHP*, permitió obtener un sistema de fácil administración y de alta calidad. Se logró como resultado con la implementación del portal web, agilizar el proceso de gestión de la información. Entre los nuevos servicios que destacan en la nueva versión del portal web se encuentran, la posibilidad de compartir los contenidos en las diferentes redes sociales, descargar documentos desde la biblioteca e intercambiar información a través del foro habilitado con tal fin. El nuevo portal permite recibir notificaciones por correo electrónico relacionadas con las noticias, brinda la capacidad de visualizar los contenidos en múltiples idiomas y el soporte necesario para ser accedido a través de dispositivos móviles. Se realizaron las pruebas de funcionalidad, usabilidad, seguridad, carga y estrés satisfactoriamente, avalando la calidad del nuevo portal.

Palabras clave: *Drupal*, Gestión de la información, Partido Comunista de Cuba, Portal web.

Índice de contenido

Introducción	1
Capítulo 1 Fundamentación teórica de la investigación	6
1.1 Introducción	6
1.2 Conceptos asociados.....	6
1.3 Análisis de portales web homólogos	6
1.4 Análisis de los sistemas de gestión de contenidos.....	14
1.5 Tecnologías y herramientas.....	18
1.6 Metodologías de desarrollo de <i>software</i>	23
1.7 Conclusiones parciales	30
Capítulo 2 Características de la propuesta de solución.....	31
2.1 Introducción	31
2.2 Características del sistema	31
2.3 Diagrama del modelo de dominio de la solución	32
2.4 Especificaciones de los requisitos del <i>software</i>	33
2.5 Arquitectura y patrones de diseño.....	39
2.6 Diagrama de despliegue del sistema	43
2.7 Conclusiones parciales	44
Capítulo 3 Implementación y pruebas de la propuesta de solución.....	45
3.1 Introducción	45
3.2 Diagrama de componentes	45
3.3 Estándares de codificación	46
3.4 Pantallas principales del portal web	51
3.5 Validación del portal web	52
3.6 Conclusiones parciales	62
Conclusiones	63
Recomendaciones	64
Referencias Bibliográficas.....	65
Bibliografía.....	69

Introducción

Un partido político es una organización que aglutina en sus filas a personas de un determinado país o región, que comparten una base en común de criterios políticos, ideas, objetivos y formas de organizar una sociedad. Se conforma bajo la base de la libre voluntad de sus integrantes y la aceptación de los estatutos y reglamentos que regulan la actividad de dicha organización tanto a lo interno como en su relación con la sociedad.

Los partidos políticos expresan los intereses sociales y clasistas de sus integrantes, y desarrollan estrategias y políticas orientadas a llevar sus ideas a una implementación social de estas. Para lograr esto parten de la necesidad de mantener una vinculación con la sociedad en que se desarrollan sus acciones proselitistas, que constituyen su base primordial para la obtención de la capacidad política que le permitirá sacar adelante sus programas estratégicos y políticos.

A nivel internacional, los partidos políticos, para la divulgación y consiguiente socialización de su información incorporan las fortalezas que confiere el uso de las facilidades ofrecidas por las Tecnologías de la Información y las Comunicaciones (TIC). Las nuevas tecnologías se utilizan para facilitar la gestión de la información vital en el funcionamiento del partido y aprovechan las virtudes del desarrollo tecnológico actual, para conectar eficientemente con la sociedad. Sus líderes, en la actualidad, tienen que responder a nuevos retos y asumir los recursos tecnológicos como un área más de acción política.

En la actualidad una de estas tecnologías es internet la cual constituye una herramienta tecnológica y poderosa de intercambio de información en el mundo, ya que posibilita el acceso a disímiles servicios, como la mensajería instantánea, las conversaciones en línea, la transmisión de archivos, el envío de correo electrónico y la web [1]. El gran impacto que suscita el uso de la internet en el mundo, no solo tecnológico sino también social y cultural, trae aparejado que los medios de comunicación, instituciones y organizaciones opten porque sus publicaciones puedan ser conocidas y accedidas a través de la red de redes.

La web es uno de los servicios con más éxito de internet por la gran cantidad de información a la que permite acceder. En la actualidad se percibe un gran avance de las tecnologías y las herramientas del desarrollo de la web. Las actuales tendencias del desarrollo de portales web persiguen entre las características

fundamentales que deben poseer, la usabilidad, la accesibilidad y a su vez, deben ser capaces de gestionar todo el ciclo de vida de los contenidos que exhiben [2]. Este procedimiento, es actualmente un elemento importante para el éxito de un portal web, debido a la necesidad de disponer de la información de forma rápida y sencilla.

Cuba le ha brindado especial atención a los avances tecnológicos, a pesar de sus dificultades tecnológicas, es por ello que el Partido Comunista de Cuba (PCC) cuenta con un portal web desde el año 1996, en el cual se muestran los estatutos, la historia, la estructura y otras informaciones relacionadas con la organización.

Sin embargo el portal vigente del PCC no emplea las facilidades ofrecidas por el desarrollo tecnológico actual. Pues no se utilizan las nuevas tecnologías para la gestión de la información en portales web, ya que esta es escrita directamente en los ficheros del portal, lo que hace este proceso muy engorroso, ya que requiere de un personal con altos conocimientos de informática para la actualización de los contenidos, impidiendo que la información se gestione de forma rápida. No posibilita a los usuarios realizar búsquedas de información y no permite la interacción informacional en ambos sentidos, de los usuarios y el sistema, convirtiéndolos en meros receptores de la información.

Se pueden citar además como elementos negativos los siguientes:

- ✓ No cuenta con un menú desplegable lo que impide aprovechar el espacio en la página principal del portal restándole usabilidad, ya que las secciones quedan parcialmente fuera de la vista del usuario haciendo necesario desplazarse por la página.
- ✓ No existe un registro de usuarios según sus diferentes niveles de acceso, siendo un elemento de suma importancia para la seguridad de la información.
- ✓ No proporciona el soporte necesario para ser accedido a través de dispositivos móviles como los *Smart Phones* (Teléfonos Inteligentes) o las *Tablets* (Tabletas), lo que afecta el acceso a la información brindada.
- ✓ Los usuarios no pueden realizar la evaluación, descarga, impresión o envíos por correo electrónico de los contenidos.

En esencia, no utiliza las facilidades tecnológicas ofrecidas por el desarrollo actual de la web lo que atenta directamente contra la inserción de las actividades partidistas en una dinámica y constante comunicación con la sociedad cubana.

Por lo planteado anteriormente se define como **problema a resolver**: ¿Cómo facilitar la gestión de la información del portal web del Partido Comunista de Cuba?

De acuerdo al presente problema, se define como **objeto de estudio**: Desarrollo de portales web.

El **objetivo general** que persigue la investigación es: Desarrollar una nueva versión del portal web del Partido Comunista de Cuba, haciendo uso de un sistema de gestión de contenidos para facilitar la gestión de la información en el sistema.

Dentro del objeto de estudio se enmarca como **campo de acción**: Desarrollo de portales web institucionales.

Desglosándose en los siguientes **objetivos específicos**:

- ✓ Construir el marco teórico conceptual de la investigación y realizar un estudio de homólogos y tecnologías necesarios para el desarrollo de portales web, que permita definir una propuesta de solución.
- ✓ Describir los requisitos funcionales y generar los artefactos necesarios para la comprensión de la solución.
- ✓ Implementar los requisitos funcionales identificados en la propuesta de solución.
- ✓ Validar los resultados del proyecto mediante pruebas de funcionalidad, usabilidad, seguridad, carga y estrés.

Se identifica como **idea a defender** que, desarrollando un nuevo portal web se facilitará la gestión de la información del portal web del Partido Comunista de Cuba.

Para dar cumplimiento a los objetivos propuestos se proponen las siguientes **tareas de la investigación**:

- ✓ Valoración de la necesidad e importancia del portal a desarrollar.
- ✓ Construcción del diseño teórico de la investigación.
- ✓ Construcción del marco teórico conceptual de la investigación.
- ✓ Valoración personal sobre el dominio de la investigación que incluya las decisiones sobre tecnologías, herramientas y lenguajes a emplear.

- ✓ Descripción de los requerimientos del sistema.
- ✓ Identificación y descripción de las entidades existentes y sus relaciones.
- ✓ Implementación de los requerimientos del sistema.
- ✓ Personalización de la interfaz de usuario.
- ✓ Realización de las pruebas de funcionalidad, usabilidad, seguridad, carga y estrés.
- ✓ Identificación de los resultados obtenidos en el trabajo.

Para el cumplimiento de los objetivos propuestos se utilizarán los siguientes **métodos de la investigación**:

Métodos empíricos:

Se utiliza el método de análisis documental: en la revisión de la literatura especializada para consultar la información necesaria en el proceso de investigación.

Métodos teóricos:

Se utiliza el método analítico – sintético: que permite consultar la bibliografía especializada en cuanto al tema abordado e identificar elementos clave que contribuyan a la solución del problema científico planteado, permite sintetizar conceptos que ayudarán a comprender la solución del problema. Este facilita el análisis de las principales metodologías, herramientas y pruebas de *software* existentes para la selección de las más adecuadas.

La modelación: Permite la representación estructural, relaciones internas y características de la solución propuesta, haciendo uso de diagramas.

Estructura Capítular

Capítulo 1. Fundamentación Teórica: Brinda un acercamiento a los principales conceptos asociados al dominio del problema que son abordados a lo largo del trabajo; así como un estado del arte de sistemas homólogos y de las principales metodologías y procedimientos usados en la actualidad para el desarrollo de sitios web informativos, que han servido de apoyo para la solución del problema planteado.

Capítulo 2. Características de la propuesta de solución: Define las principales características que debe cumplir el sistema a desarrollar. Se muestra el diagrama de componentes y el de despliegue. Son enumerados y descritos los requisitos no funcionales y los requisitos funcionales, estos últimos son descritos mediante historias de usuario.

Capítulo 3. Implementación y pruebas de la propuesta de solución: En este capítulo se muestran y se establecen los estándares de codificación que se tuvieron en cuenta para el desarrollo del portal web. Se describen las pruebas realizadas para la validación del portal web y se exponen los resultados obtenidos.

Capítulo 1 Fundamentación teórica de la investigación

1.1 Introducción

En el presente capítulo se engloban aspectos relacionados con el objeto de estudio definido para el problema planteado. El análisis de algunas metodologías, procedimientos, herramientas existentes para el desarrollo de sitios web y la observación de sitios homólogos que brindan información política referente a diversos partidos en el mundo; permitirá la selección de las tecnologías adecuadas para el desarrollo del portal web del PCC y contar con un análisis de sistemas existentes que realizan funcionalidades similares.

1.2 Conceptos asociados

1.2.1 Sitio web

Un sitio Web es un conjunto organizado y coherente de páginas web (generalmente archivos en formato *HTML (HyperText Markup Language* o Lenguaje de Marcas de Hipertexto), *PHP (Hypertext Preprocessor* o Preprocesador de Hipertexto), etc.) y objetos (gráficos, animaciones, sonidos, etc.). Su temática puede ser muy variada. A través de un sitio web se puede ofrecer, informar, publicitar o vender contenidos, productos y servicios al resto del mundo. Para que un sitio web pueda ser visitado desde cualquier parte del mundo por otras personas es necesario que se encuentre alojado en un servidor. Se trata de una computadora conectada a la *WWW (World Wide Web)* con espacio en disco y conectividad suficiente para albergar sitios y servirlos al resto de la comunidad de usuarios de internet a través de direcciones ip o nombres de dominio [3].

Es una localización en la *WWW* que contiene documentos (páginas web) organizados jerárquicamente, que está típicamente dedicado a algún tema particular o propósito específico; los que pueden ser muy variados, ya que los sitios pueden ser utilizados de diferentes formas por distintas instituciones; ya sea para negocios, divulgar información o prestar servicios, lo que permite lograr como ventajas un alcance mundial y estar disponibles 24 horas.

1.3 Análisis de portales web homólogos

A continuación se expone un análisis de los portales web de algunos partidos políticos, para obtener características comunes en cada uno de ellos y luego emplearlas en la propuesta de solución.

1.3.1 Portal web del Partido Comunista de Vietnam (PCV)

El portal web del Partido Comunista de Vietnam (PCV), es el portal que representa de cara a internet al PCV. Este partido comunista de tendencia marxista-leninista fue fundado por *Ho Chi Minh* y otros exiliados que vivían en China el 3 de Febrero de 1930.

El portal web del PCV exhibe una gama de colores en la que predomina el rojo, así como un logo animado que representa al PCV. Muestra su información organizada por secciones y separadas por bloques para delimitar los contenidos.

Cuenta con 13 secciones principales:

- ✓ *About Us* (Sobre Nosotros): muestra información sobre las responsabilidades que tiene el PCV.
- ✓ *News and Events* (Noticias y Eventos): exhibe todas las noticias que están publicadas en el portal web.
- ✓ *Activities of the Party and State* (Actividades del Partido y el Estado): muestra noticias relacionadas con actividades que ejecutan el partido y el estado.
- ✓ *Economics* (Económicas): presenta noticias de temáticas referentes a la economía de Vietnam.
- ✓ *Social Affairs* (Asuntos Sociales): brinda información sobre asuntos sociales del país.
- ✓ *Culture* (Cultura): muestra noticias relacionadas con los eventos culturales.
- ✓ *Science-Education* (Ciencia-Educación): brinda publicaciones sobre los avances y eventos de la ciencia y la educación en Vietnam.
- ✓ *Foreign Affairs* (Asuntos Exteriores): exhibe información sobre relaciones y acontecimientos con otros países del mundo.
- ✓ *Overseas Vietnamese* (Vietnamitas en el Extranjero): ofrece noticias de eventos en el extranjero de cualquier ámbito en los que vietnamitas participan.
- ✓ *Sports* (Deportes): brinda noticias sobre el deporte.
- ✓ *Tourism* (Turismo): muestra publicaciones sobre el turismo.
- ✓ *World: Events and Comments* (Mundo: Eventos y Comentarios): brinda información de hechos y comentarios de ámbito internacional.
- ✓ *Vietnam in foreigners eyes* (Vietnam en los ojos de los extranjeros): muestra noticias sobre el enfoque del extranjero hacia Vietnam en disímiles áreas.

Las últimas publicaciones asociadas a sus diferentes secciones como noticias y eventos, actividades del partido y el estado, económicas, asuntos sociales, galerías de imagen, video y audio, se pueden encontrar fácilmente en la página principal. También en esta se muestran publicaciones que brindan documentos y archivos relacionados con el partido.

El portal web del PCV brinda servicios como internacionalización, impresión y búsqueda de los contenidos a través de palabras clave. Presenta enlaces a sitios de interés, el mapa del sitio para un mayor conocimiento de la estructura del portal web y cuenta con un foro de debates.

Sin embargo no cuenta con espacios en las redes sociales y canales como *Facebook*, *Twitter*, *RSS (Really Simple Syndication* o *Sindicación Realmente Simple*) y otros medios de acceso para interactuar y socializar sus contenidos en internet. No brinda servicio de suscripción a boletines, tampoco página con información de contacto y no proporciona el soporte necesario para ser accedido a través de dispositivos móviles como los Teléfonos Inteligentes o Tabletás.

1.3.2 Portal web del Partido Comunista de China (PCCh)

El PCCh es el partido político que gobierna la República Popular China. Con más de 71 millones de miembros, es una de las mayores organizaciones políticas del mundo, tras su fundación en 1921, el PCCh se convirtió en una de las formaciones políticas más influyentes de China.

El portal web del PCCh exhibe una gama de colores en la que predomina el rojo y presenta un logo que identifica al PCCh. Muestra su información organizada por secciones y separadas por bloques para delimitar los contenidos.

Cuenta con 6 secciones principales:

- ✓ *Latest News* (Últimas Noticias): muestra todas las noticias publicadas en el portal web organizadas por fecha.
- ✓ *Leaders Activities* (Actividades de Líderes): ofrece noticias referentes a las actividades donde han participado líderes del partido.
- ✓ *Achievements* (Logros): brinda información sobre los logros más importantes alcanzados en el país.

- ✓ *Comments* (Comentarios): muestra opiniones escritas por moderadores del portal web relacionadas con diferentes temáticas.
- ✓ *Democracy* (Democracia): brinda artículos relacionados con la democracia.
- ✓ *Mass Line* (Línea de Masas): trata sobre las líneas de masas del partido.
- ✓ *Photos* (Fotos): muestra las galerías de imágenes.

Las últimas publicaciones asociadas a sus diferentes secciones como últimas noticias, actividades de líderes, democracia y fotos, se pueden encontrar fácilmente en la página principal. También se muestra información relacionada con el partido como documentos, archivos, estructura y congresos.

El portal web del PCCh brinda servicios como internacionalización y la búsqueda de los contenidos a través de palabras clave. Presenta enlaces a sitios de interés y cuenta con la opción de compartir en las redes sociales y comentar los contenidos. Brinda el mapa del sitio para un mayor conocimiento de la estructura del portal web y página con información de contacto.

Sin embargo no brinda servicios como suscripción a boletines e impresión. No proporciona el soporte necesario para ser accedido a través de dispositivos móviles como las Tabletas y no posee espacios en las redes sociales y canales como *Facebook*, *Twitter* y *RSS*.

1.3.3 Portal web del Partido Socialista Unido de Venezuela (PSUV)

El PSUV es un partido político venezolano donde se fusionan algunas de las fuerzas políticas y sociales que apoyan a la Revolución Bolivariana impulsada por Hugo Chávez. Durante su mandato presidencial Chávez se dio a la tarea de convencer a los diversos partidos políticos que lo apoyaban para que se disolvieran e ingresaran en el PSUV.

El portal web del PSUV exhibe una gama de colores en la que predomina el color rojo, así como un logo que representa al PSUV. Muestra su información organizada por secciones y separadas por bloques para delimitar los contenidos.

Cuenta con 13 secciones principales:

- ✓ Notas de Retaguardia: muestra diferentes publicaciones hechas por Hugo Chávez referente a diferentes temáticas.
- ✓ Líneas de Chávez: brinda reflexiones realizadas por Hugo Chávez referente a diferentes temáticas.
- ✓ Opinión: muestra opiniones escritas por moderadores del portal relacionadas con diversos temas.
- ✓ Comunicados: exhibe comunicados pertinentes con el partido y de otros ámbitos.
- ✓ Noticias: presenta todas las noticias del portal.
- ✓ Memorias: muestra publicaciones ligadas a fechas de conmemoraciones y eventos importantes.
- ✓ PSUV: brinda el acceso a información como el himno, estatutos, gobernadores, alcaldes y diputados del partido.
- ✓ JPSUV: es un enlace al portal web de la Juventud del Partido Socialista Unido de Venezuela (JPSUV).
- ✓ Gráfica: muestra imágenes relacionadas con el partido, el pueblo, campañas realizadas y las elecciones regionales.
- ✓ Audios: permite escuchar canciones utilizadas en los distintos eventos de ámbito político.
- ✓ Galería: permite ver las fotos de los eventos políticos y otras de ámbito diferente.
- ✓ Constitución: exhibe la documentación de la Constitución de la República Bolivariana de Venezuela.
- ✓ Biblioteca: brinda bibliografías de temas políticos, militares y medio ambiente.

Las últimas publicaciones asociadas a sus diferentes secciones como noticias, reflexiones de Fidel, líneas de Chávez, opiniones, entrevistas y galería se pueden encontrar fácilmente en la página principal. También se muestra información relacionada con el partido como documentos, estructura, himno, estatutos y su historia.

El portal web del PSUV brinda servicios como la búsqueda de los contenidos a través de palabras clave y suscripción a boletines. Presenta enlaces a sitios de interés, cuenta con un foro de debate y permite comentar los contenidos. Posee espacios en las redes sociales y canales como *Facebook*, *Twitter* y *RSS*.

Sin embargo no cuenta con la opción de compartir los contenidos en las redes sociales, tampoco con servicios de internacionalización y de impresión. No brinda el mapa del sitio y no proporciona el soporte necesario para ser accedido a través de dispositivos móviles como los teléfonos inteligentes.

1.3.4 Portal web del Partido Popular de España (PP)

El PP es un partido político español conservador, liberal, situado en la centro-derecha o derecha fundado en 1989, cuando adoptó su denominación actual, que sustituyó a la de Alianza Popular. Cuenta según sus propias estimaciones, con más de 865000 afiliados divididos entre militantes (que pagan cuotas) y simpatizantes (que no la pagan).

El portal web del PP exhibe una gama de colores en la que predomina el color azul, así como un logo que representa al PP. Muestra su información organizada por secciones y separadas por bloques para delimitar los contenidos.

Cuenta con 10 secciones principales:

- ✓ Conócenos: se divide en subsecciones en las cuales se muestran diferentes informaciones relacionadas directamente con el partido como su historia, programas, sedes, fundaciones y estructura.
- ✓ Temas: se divide en subsecciones en las que se muestran publicaciones relacionadas con cada una como noticias, documentos, videos e imágenes; donde se exponen todas las acciones que realiza el PP para resolver problemas relacionados con distintos temas como empleo, economía, la igualdad de la mujer y corrupción en España.
- ✓ Actualidad: se divide en subsecciones que exhiben las noticias, videos, imágenes y documentos del portal web. También se muestra un calendario de actividades programadas.
- ✓ Participa: se divide en subsecciones en las cuales se indica cómo colaborar con el partido.
- ✓ Congreso: brinda información sobre la actividad del congreso de los diputados.
- ✓ Senado: muestra información sobre la actividad en el senado.
- ✓ Gobierno: exhibe información relacionada con las medidas del gobierno de Mariano Rajoy.
- ✓ Europa: ofrece información referente a la labor del PP en Europa.
- ✓ Medios: contiene información relacionada con la actualidad del PP en los medios de comunicación.

- ✓ **Contacta:** brinda toda la información de contacto necesaria y cuenta con un formulario el cual permite enviar sugerencias.

Las últimas publicaciones asociadas a sus diferentes secciones y subsecciones como noticias, empleo, economía, transparencia y regeneración, agenda, imágenes y videos se pueden encontrar fácilmente en la página principal. También en el portal web se muestra información como documentos, historia, estructura y programas del partido.

El portal web del PP brinda servicios como la búsqueda a través de palabras clave. Brinda el mapa del sitio para un mayor conocimiento de la estructura del portal web y página con información de contacto. Presenta enlaces a sitios de interés y proporciona el soporte necesario para ser accedido a través de dispositivos móviles como los Teléfonos Inteligentes. Posee espacios en las redes sociales y canales como *Facebook*, *Twitter* y *RSS*.

Sin embargo no brinda servicios de internacionalización, impresión y suscripción a boletines. No permite comentar los contenidos y tampoco compartirlos en las redes sociales y además no posee foro de debates.

1.3.5 Portal web del Partido del Movimiento Democrático Brasileño (PMDB)

El PMDB, que nació del MDB, tuvo su fundación el 24 de marzo de 1966, con el expediente en la Corte Electoral. Fue uno de los resultados de la extinción de los 13 partidos impuesta por el Acto Institucional Número Dos y la instalación de bipartidismo poco después.

El portal web del PMDB exhibe una gama de colores en la que predominan el color rojo y el blanco, así como un logo que representa al PMDB. Muestra su información organizada por secciones y separadas por bloques para delimitar los contenidos.

Cuenta con 8 secciones principales:

- ✓ **Institucional:** se divide en diferentes subsecciones en las cuales se muestran publicaciones relacionadas con la historia, los estatutos, código de ética, figuras históricas, programa del PMDB y otras informaciones relacionadas directamente con dicho partido.

- ✓ Núcleos: separada en subsecciones en las que se brinda información acerca de los distintos núcleos con los que cuenta el partido y las características de cada uno.
- ✓ Qué Hace: es una sección muy importante dividida también en subsecciones las cuales representan la estructura del partido y en cada una de ellas se brinda información sobre los diputados, senadores, ministros, gobernadores, vice-gobernador y vice-presidente de la república.
- ✓ Noticias: dividida también en subsecciones en las cuales se muestran todas las noticias, los artículos, entrevistas y las multimedia del portal web.
- ✓ Contacto: muestra informaciones de contacto como el número y el correo electrónico de la cámara de diputados.
- ✓ Link: brinda enlaces a sitios de interés.
- ✓ PMDB por los estados: brinda un mapa interactivo en el que se muestra por estados todas las sucursales del partido e información referentes a ellas.
- ✓ Contáctenos: muestra un formulario el cual permite enviar sugerencias y opiniones al PMDB.

Las últimas publicaciones asociadas a sus diferentes secciones y subsecciones como noticias, artículos, calendario, entrevistas, galería de imágenes, audio y video, se pueden encontrar fácilmente en la página principal.

El portal web del PMDB brinda servicios como la impresión y búsqueda de los contenidos a través de palabras clave. Presenta enlaces a sitios de interés y proporciona el soporte necesario para ser accedido a través de dispositivos móviles como las Tablet. Brinda un boletín del sitio en formato PDF el cual se puede descargar y cuenta con la opción de comentar los contenidos. Posee espacios en las redes sociales y canales como *Facebook*, *Twitter* y *RSS*.

Sin embargo no brinda servicios de internacionalización y suscripción a boletines. No cuenta con la opción de compartir los contenidos en las redes sociales. No posee foro de debates y tampoco mapa del sitio.

Resultados obtenidos del análisis de los portales web homólogos

Con este estudio se determina que la mayoría de los portales web brindan información referente a la institución como su historia, tareas cruciales, personalidades históricas, funciones, compromisos, responsabilidades y estructura. Todos brindan una búsqueda simple a través de palabras clave, algunos

brindan foros de debates y servicios como envío de boletines, multimedia, internacionalización, impresión y muestran el mapa del sitio. Muestran su información organizada por secciones y separadas por bloques para delimitar los contenidos. Tienen un manejo de contraste donde hacen uso de una gama de colores agradables para el usuario y en consonancia con su identidad. En su mayoría poseen espacios en las redes sociales y canales como *Facebook*, *Twitter* y *RSS*, y permiten compartir los contenidos. Algunos proporcionan el soporte necesario para ser accedidos a través de dispositivos móviles como los Teléfonos Inteligentes y la opción de comentar los contenidos.

1.4 Análisis de los sistemas de gestión de contenidos

Los Sistemas de Gestión de Contenidos (*Content Management Systems* o *CMS*) se utilizan principalmente para facilitar la implementación de los procesos de gestión. Una de las características de estos sistemas, que además constituye una ventaja con respecto a su uso, es el hecho de posibilitar la inmediatez de la publicación y edición de los contenidos [4].

Ventajas de los sistemas de gestión de contenidos:

- Capacidad de manejar eficientemente gran cantidad de páginas web.
- Trabajar en un ambiente de páginas web interactivas, es decir, que se generan según las peticiones de los usuarios.
- Controlar el acceso de los usuarios al sistema, no sólo mediante su contraseña, sino también mediante permisos asignados a cada uno.
- Orden en el sistema, al existir la posibilidad de asignar, por parte de la herramienta, un mismo estilo a todas las páginas generadas.

Existe una amplia variedad de *CMS* que según el tipo de licencia son clasificados en comerciales (*CoreMedia CMS*, *PipePS*, *NUKE ET*) o de código abierto (*Joomla*, *Drupal*, *WordPress*) [5]. Pero el análisis se centra en los de código abierto ya que es la mejor elección para el desarrollo de la propuesta de solución, pues no es costoso y brinda acceso al código fuente. A continuación se analizan las características de algunos de estos *CMS*.

1.4.1 Sistema de gestión de contenidos *WordPress*

WordPress es un *CMS* de gran sencillez de uso hasta el punto donde solo puede usarse para publicar artículos, por lo que es muy utilizado para la creación de *Blog*. Aclarar que *WordPress* (núcleo) no es un *CMS* propiamente dicho, aunque puede evolucionar hasta llegar a serlo, y también puede ser extensible utilizando *plugins*¹ adicionales. 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. Para su desarrollo utiliza *PHP* y *MySQL* y es uno de los *CMS* más populares junto a *Joomla* y *Drupal*. Debido a que los blogs son los sistemas más susceptibles de recibir *spam*², *Wordpress* posee un *plugin* (complemento) llamado "*Akismet*" que identifica y frena la mayoría de los comentarios y *trackbacks*³ que son *spam* [6].

Wordpress presenta algunas desventajas, por ejemplo: no permite modificar el código del sitio fácilmente. Dependiendo de la cantidad y frecuencia de las publicaciones la administración de *blog* puede dificultarse, si no se realiza una revisión periódica. Por ser un espacio de acceso público pueden recibirse comentarios no deseados que no se encuentren relacionados con las temáticas. El acceso a *blog* debe hacerse, necesariamente utilizando un navegador. Si no se conoce con certeza la dirección del *blog*, la búsqueda se torna difícil.

1.4.2 Sistema de gestión de contenidos *Drupal*

Drupal es un sistema de gestión de contenido de código abierto, con licencia *GNU/GPL*⁴, escrito en *PHP*, desarrollado y mantenido por una activa comunidad de usuarios. Se destaca por la calidad de su código y de las páginas generadas, el respeto de los estándares de la web, un énfasis especial en la usabilidad y consistencia de todo el sistema. El diseño de *Drupal* es especialmente idóneo para construir y gestionar comunidades en internet. Si bien las características básicas de *Drupal* son suficientes como para lanzar un sitio completo, incluye: administración de usuarios, vocabulario, comentario y sindicación, pero su

¹ Complemento, una aplicación que se relaciona con otra para aportarle una función nueva y generalmente muy específica.

² Correo electrónico o publicidad engañosa no solicitada que se envía a un gran número de destinatarios.

³ Enlace inverso que permite conocer qué enlaces apuntan hacia un determinado artículo.

⁴ Siglas del inglés *GNU* (*General Public License*), traducido al español como Licencia Pública General de *GNU*.

potencialidad reside en su estructura modular que permite ser extendida y configurada por el administrador para distintas funciones por medio de módulos de fácil instalación [7].

Gracias a esta estructura modular se pueden agregar nuevas funciones a módulos ya existentes y crear nuevos módulos sin afectar los ya creados. Estos módulos son los encargados del funcionamiento del sistema y separan la interfaz gráfica de la información. La desventaja que presenta *Drupal* es la alta curva de aprendizaje y la gran cantidad de módulos, que hace difícil la selección de los más eficientes.

1.4.3 Sistema de gestión de contenidos *Joomla*

Joomla es uno de los sistemas gestores de contenidos de código abierto más popular que permite crear sitios web de alta interactividad, profesionalidad y eficiencia. La administración de *Joomla* está enteramente basada en la gestión *online* de contenidos. Se dice "gestión *online*" porque todas las acciones que realizan los administradores de sitios *Joomla*, ya sea para modificar, agregar, o eliminar contenidos se realizan exclusivamente mediante un navegador (*browser*) conectado a internet, es decir, a través del protocolo *HTTP* (Protocolo de Transferencia de Hipertexto).

Joomla tiene una gran comunidad de usuarios y también toda la documentación para crear diferentes aplicaciones. Además, es posible realizar casi cualquier sitio con muy pocos conocimientos: comercio electrónico, revistas *online*, *intranets*, redes sociales, etc. Realiza un gran trabajo gestionando el contenido necesario para que un sitio web funcione. Para muchas personas, el verdadero potencial de dicho CMS reside en la arquitectura de la aplicación, que posibilita que miles de desarrolladores en el mundo puedan crear potentes *add-ons*⁵ y extensiones. Algunas de las numerosas extensiones disponibles son: gestores de documentos, generadores de formularios dinámicos, galerías de imágenes multimedia, motores de comercio y venta electrónica, directorios de empresas u organizaciones, *software* de foros y *chats*, calendarios, *software* para *blogs*, servicios de directorio, boletines de noticias, herramientas de registro de datos, sistemas de publicación de anuncios, servicios de suscripción. Está programado en lenguaje *PHP* y *SQL*. Utiliza bases de datos relacionales, más específicamente *MySQL* y al ser *Joomla* una aplicación web, funciona obviamente en servidores de páginas web (*HTTP Servers*) [8].

⁵ Complemento, una aplicación que se relaciona con otra para aportarle una función nueva y generalmente muy específica.

Sin embargo *Joomla* posee una pobre gestión de usuario, siendo este elemento necesario para la propuesta de solución. Las opciones para crear y personalizar nuevos rangos así como permisos de usuarios muchas veces se hacen insuficiente.

1.4.4 Consideraciones de los sistemas de gestión de contenidos

Se escoge *Drupal* en su versión 7.35 pues ofrece todas las prestaciones y funcionalidades necesarias para gestionar los contenidos de un portal web. Permite gracias a su estructura modular añadir nuevas funcionalidades sin afectar las ya existentes y sin afectar el sistema. También proporciona gestión de usuarios, clasificación y edición de los contenidos, y posee un potente sistema de seguridad basado en roles, que permite controlar los diferentes módulos según los permisos que el administrador le defina a cada rol.

Otras ventajas de *Drupal*:

- **Ayuda on-line:** Un robusto sistema de ayuda online y páginas de ayuda para los módulos del núcleo.
- **Búsqueda:** Todo el contenido en *Drupal* es totalmente indexado en tiempo real y se puede consultar en cualquier momento.
- **Autenticación de usuarios:** Los usuarios se pueden registrar e iniciar sesión de forma local.
- **Permisos basados en roles:** Los administradores de *Drupal* no tienen que establecer permisos para cada usuario, debido a que pueden asignar permisos a un rol y agrupar los usuarios por roles.
- **Plantillas:** El sistema de temas de *Drupal* separa el contenido de la presentación permitiendo controlar o cambiar fácilmente el aspecto del sitio web.
- **Sindicación del contenido:** Exporta el contenido en formato *RSS*.
- **Independencia de la base de datos:** Incorpora una capa de abstracción de base de datos.
- **Multiplataforma:** *Drupal* ha sido diseñado desde el principio para ser multiplataforma.
- **Múltiples idiomas y localización:** *Drupal* está pensado para una audiencia internacional y proporciona opciones para crear un portal multilingüe. Todo el texto puede ser fácilmente traducido utilizando una interfaz web, importando traducciones existentes o integrando otras herramientas de traducción.
- **Administración vía web:** La administración y configuración se puede realizar enteramente con un navegador y no precisa de ningún *software* adicional.

- **Sistema de *cache*:** El mecanismo de *cache* elimina consultas a la base de datos incrementando el rendimiento y reduciendo la carga del servidor.

1.5 Tecnologías y herramientas

En la realización de un proyecto es imprescindible la etapa donde se definen las tecnologías y herramientas a utilizar así como la versión de cada una de ellas que será empleada. Por esta razón se determinó el empleo de las herramientas y tecnologías que a continuación se enuncian, estas están limitadas por el *CMS* seleccionado:

1.5.1 Lenguaje del lado del servidor

Se clasifica así al lenguaje de programación en la arquitectura cliente servidor. Un lenguaje del lado del servidor es aquel que se ejecuta en el servidor web del cual los usuarios solo obtienen el beneficio del procesamiento de la información [9].

Pre-Procesador de Hipertexto (*PHP*)

PHP es un lenguaje de programación interpretado, diseñado originalmente para la creación de páginas web dinámicas [10]. Es conocido como una tecnología de código abierto que resulta muy útil para diseñar de forma rápida y eficaz aplicaciones web dirigidas a bases de datos. Su interpretación y ejecución se realizan en el servidor en el cual se encuentra almacenada la página y el cliente sólo recibe el resultado de la ejecución. Entre sus ventajas más significativas se presentan las siguientes:

- Multiplataforma.
- Completamente orientado a la web.
- Capacidad de conexión con la mayoría de los motores de base de datos que se utilizan en la actualidad.
- Permite las técnicas de programación orientada a objetos.
- Manejo de excepciones (desde *PHP5*).

Derivado del uso del *CMS* empleado se hace uso de *PHP5*.

1.5.2 Lenguajes del lado del cliente

Los lenguajes del lado del cliente son totalmente independientes del servidor. Permiten que la página web pueda ser albergada en cualquier sitio [11].

Lenguaje marcado de hipertexto (*HTML*)

HTML es un lenguaje que se utiliza fundamentalmente en el desarrollo de páginas web. *HTML* son las siglas de *HiperText Markup Language* (Lenguaje de Marcación de Hipertexto) es un lenguaje que se utiliza comúnmente para establecer la estructura y contenido de un sitio web, tanto de texto, objetos e imágenes. Los archivos desarrollados en *HTML* usan la extensión .htm o .html. Funciona por medio de etiquetas que describen la apariencia o función del texto enmarcado [12]. Algunas de estas etiquetas son < b > que rodea al texto en negrita, < i > pone el texto en cursiva y < u > que lo subraya.

Lenguaje de marcado (*XML*)

La sigla *XML* es la abreviación de la expresión "*Extensible Markup Language*", lo que podría traducirse como lenguaje de marcas extensible, que permite definir la gramática de lenguajes específicos para estructurar documentos grandes. A diferencia de otros lenguajes da soporte a bases de datos, siendo útil cuando varias aplicaciones se deben comunicar entre sí o integrar información. No es un lenguaje en particular, sino una manera de definir lenguajes para diferentes necesidades [13].

JavaScript

JavaScript es un lenguaje de programación que se utiliza principalmente para crear páginas web dinámicas. Una página web dinámica es aquella que incorpora efectos como aparición y desaparición de texto, animaciones, acciones que se activan al pulsar botones u otros elementos y ventanas con mensajes de aviso al usuario. Técnicamente, *JavaScript* es un lenguaje de programación interpretado, por lo que no es necesario compilar los programas para ejecutarlos. En otras palabras, los programas escritos con *JavaScript* se pueden probar directamente en cualquier navegador sin necesidad de procesos intermedios. [14].

Hojas de estilo en cascada (*CSS*)

Cascading Style Sheets (*CSS*) conocido en español como Hojas de Estilo en Cascada. Es un mecanismo simple que nos permite controlar la apariencia de una página web. Describe cómo se muestra un documento en la pantalla, cómo se va a imprimir o incluso cómo va a ser pronunciada la información presente en ese documento a través de un dispositivo de lectura.

Esta forma de descripción de estilos le permite a los desarrolladores web controlar el estilo y formato de múltiples páginas web al mismo tiempo. Cualquier cambio en el estilo marcado para un elemento en el *CSS* afectará a todas las páginas vinculadas a ese *CSS* en las que aparezca ese elemento [15].

1.5.3 Lenguaje de modelado

Un lenguaje de modelado establece un conjunto de notaciones y diagramas estándar para modelar sistemas describiendo la semántica esencial de estos diagramas y los símbolos en ellos utilizados. Es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema de *software*.

Lenguaje Unificado de Modelado (*UML*)

El lenguaje unificado de diagrama o notación *UML* (por sus siglas en inglés, *Unified Modeling Language*) sirve para especificar, visualizar y documentar esquemas de sistemas de *software*. *UML* no es un método de desarrollo, lo que significa que no sirve para determinar qué hacer en primer lugar o cómo diseñar el sistema, sino que simplemente ayuda a visualizar el diseño y hacerlo más accesible para otros [16]. Implementa un lenguaje de modelado común para todos los programadores mediante una documentación que cualquier desarrollador con conocimientos de *UML* pueda entender. Su utilización es independiente del lenguaje de programación y de las características de los proyectos, pues ha sido diseñado para modelar los más variados tipos de negocio.

1.5.4 Sistema gestor de base de datos

Un sistema gestor de base de datos (en inglés *DBMS: Data Base Management System*) es un sistema de *software* que permite la definición de bases de datos; así como la elección de las estructuras de datos necesarias para el almacenamiento y búsqueda, ya sea de forma interactiva o a través de un lenguaje de programación. Brinda facilidades eficientes y un grupo de funciones con el objetivo de garantizar la confidencialidad, calidad, seguridad e integridad de los datos que contienen, así como un acceso fácil y eficiente a estos [17].

PostgreSQL

PostgreSQL es un sistema de gestión de bases de datos objeto-relacional, distribuido bajo licencia *BSD*⁶ y con su código fuente disponible libremente. Es el sistema de gestión de bases de datos de código abierto más potente del mercado y en sus últimas versiones [18]. Se destaca por ejecutar consultas complejas y

⁶ Licencia de software libre otorgada principalmente para los sistemas *BSD* (*Berkeley Software Distribution*).

sobre vistas, subconsultas y *joins*⁷ de gran tamaño. Mediante un sistema denominado *MVCC* (Acceso Concurrente Multiversión) permite que mientras un proceso escribe en una tabla, otros accedan a la misma tabla sin necesidad de bloqueos. Posee una amplia variedad de tipos nativos (números de precisión arbitraria, texto de largo ilimitado, figuras geométricas). Tiene gran soporte para vistas, procedimientos ubicados en el servidor, transacciones, almacenamiento de objetos de tamaño considerable y características orientadas a objetos.

Se hace uso de *PostgreSQL* en su versión 9.1.

PgAdmin

PgAdmin 1.14.0 es una herramienta para la administración de la base de datos, de código abierto y cuenta con una interfaz gráfica que soporta todas las características de *PostgreSQL*. Se encuentra disponible en más de una docena de lenguajes y para varios sistemas operativos, incluyendo *Windows*, *Linux*, *FreeBSD* y *Mac OSX*. Esta herramienta simplifica en gran medida la administración.

1.5.5 Entorno integrado de desarrollo

NetBeans

Es una plataforma integrada de desarrollo bajo licencia *GPL* y de código abierto. Esta herramienta tiene la finalidad de permitirle a los desarrolladores escribir, compilar, depurar y ejecutar programas. En sus inicios fue realizado para el lenguaje de programación *Java*, pero por sus características se extendió a otros lenguajes como *PHP*, *JavaScript* y *Python*. Existe además un número importante de módulos para extenderlo [19]. El trabajo con el lenguaje *PHP* en el *NetBeans* posee considerables ventajas, ya que facilita el trabajo con el completamiento de código, cierra automáticamente los corchetes, comillas y paréntesis, muestra la opción de autocompletar código con variables de otros ficheros del mismo proyecto y facilita la búsqueda de variables.

Se hace uso del *NetBeans* en su versión 8.0.

⁷ Sentencia en SQL que permite combinar registros de dos o más tablas en una base de datos relacional.

1.5.6 Servidor de aplicaciones web

Un servidor web es un programa que permite crear un servidor *HTTP* en un ordenador. Con el modelo cliente/servidor y el protocolo de transferencia de hipertexto de internet, proporciona a los usuarios archivos que conforman las páginas web (textos complejos con enlaces, figuras, formularios, botones y objetos incrustados como animaciones o reproductores de música) [20].

Servidor web Apache

Es un servidor web *HTTP* de código abierto. Puede ser usado en varios sistemas operativos, lo que lo hace prácticamente universal. Es usado para tareas donde el contenido necesita ser puesto a disposición en una forma segura y confiable. *Apache* se distribuye como *software* libre de código abierto, modular multiplataforma, extensible, popular (fácil de conseguir ayuda/suporte) y gratuito [21]. Está diseñado para ser un servidor web potente y flexible que pueda funcionar en la más amplia variedad de plataformas y entornos.

Se hace uso del servidor web *Apache 2.2*.

1.5.7 Herramienta case

Visual Paradigm

Como herramienta *case* (caso) se hizo uso de *Visual Paradigm*, esta es una herramienta que soporta el modelado mediante *UML* y proporciona asistencia a los analistas, ingenieros de *software* y desarrolladores, durante todos los pasos del ciclo de vida de desarrollo de un *software* [22]. Además permite dibujar todos los tipos de diagramas de clases, código inverso, generar código desde diagramas y generar documentación. Entre sus características se destacan las siguientes:

- Soporta aplicaciones web.
- Genera informes usables en la generación de documentación.
- Facilita la importación y exportación de ficheros.
- Fácil de instalar y actualizar.
- Compatibilidad entre ediciones.

Se hace uso del *Visual Paradigm* en su versión 8.0.

1.6 Metodologías de desarrollo de *software*

En el mundo del desarrollo de *software*, sus programadores tienden a hablar de cuestiones técnicas, lenguajes de programación y entornos de desarrollo, pero no tienen en cuenta, cómo se debe hacer el *software*. Esta se ve reflejada a través de las metodologías de desarrollo de *software*, que son un conjunto de procedimientos, técnicas y ayudas a la documentación para el desarrollo de productos de *software* eficientes. Las metodologías de desarrollo de *software* surgen ante la necesidad de utilizar una serie de procedimientos, técnicas herramientas, y soporte documental a la hora de desarrollar un *software*. Dichas metodologías tienen como objetivo guiar a los desarrolladores en todo el proceso de desarrollo [23].

Están estructuradas en fases y engloban todo el ciclo de vida del desarrollo de un producto de *software* con calidad. Son clasificadas en dos grupos, las metodologías tradicionales o pesadas, entre las que se encuentran: *RUP* (*Rational Unified Process*) y por otra parte las metodologías ágiles como son: *XP* (*Extreme Programing*), *AUP* (*Agile Unified Process*) y *OpenUp* (*Open Unified Process*).

A continuación se realiza un análisis de algunas de estas metodologías (*RUP*, *XP*, *OpenUp* y *AUP*) con el objetivo de seleccionar la adecuada para el desarrollo de la propuesta de solución.

1.6.1 *Rational Unified Process (RUP)*

RUP (*Rational Unified Process* o Proceso Unificado de Racional) es una de las metodologías conocidas como tradicionales o pesadas. Consiste en un conjunto de actividades necesarias para transformar los requerimientos del usuario en un sistema de *software*. En la preparación de todos los planos del sistema usa el *UML*. Los aspectos distintivos de *RUP* están capturados en tres conceptos clave: dirigido por casos de uso (*use-case driven*); centrado en la arquitectura (*architecture-centric*); iterativo e incremental [24].

La metodología *RUP* divide en 4 fases el desarrollo del *software*. Cada una de las fases se desarrolla mediante un ciclo de iteraciones como muestra la **Figura 1** [24].

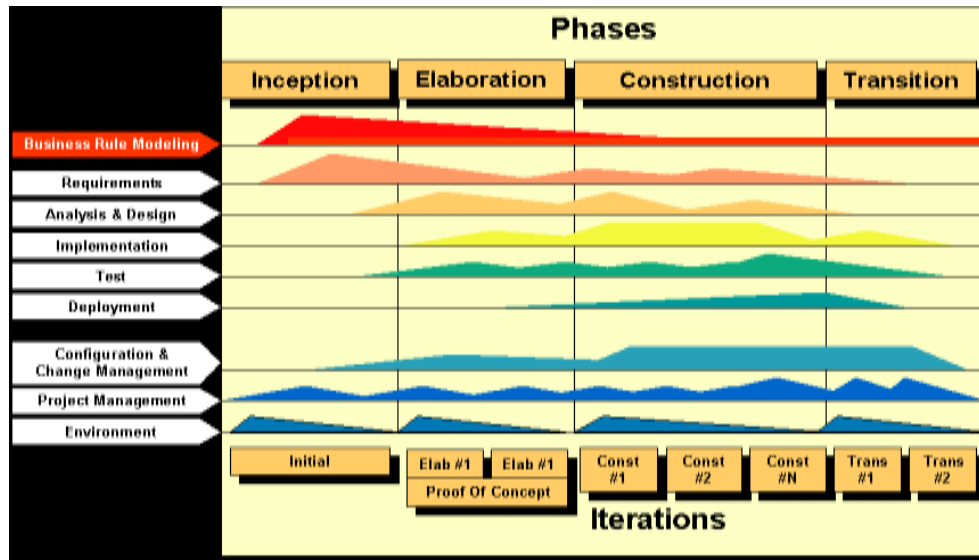


Figura 1. Fases de iteraciones de la metodología RUP

La metodología RUP es recomendada para guiar el desarrollo de proyectos complejos, de larga duración, y requiere de un equipo de trabajo capaz de administrar un proceso complejo en varias etapas. En proyectos pequeños, es posible que no se puedan cubrir los costos de dedicación del equipo de profesionales necesarios. Además no permite flexibilidad en los cambios que puedan surgir en el desarrollo del proyecto.

1.6.2 eXtreme Programing (XP)

XP (eXtreme Programing o Programación Extrema) es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en el desarrollo de software. Consiste básicamente en ajustarse estrictamente a una serie de reglas que se centran en las necesidades del cliente para lograr un producto de buena calidad en poco tiempo.

Promueve el trabajo en equipo y es apropiada para los proyectos con requisitos imprecisos, muy cambiantes y con un riesgo técnico excesivo. Entrega el sistema a los clientes tan pronto como sea posible e implementa cambios que se sugieran. XP desarrolla 4 actividades que guían el desarrollo: codificar, hacer pruebas, escuchar y diseñar como se muestra en la **Figura 2** [25].

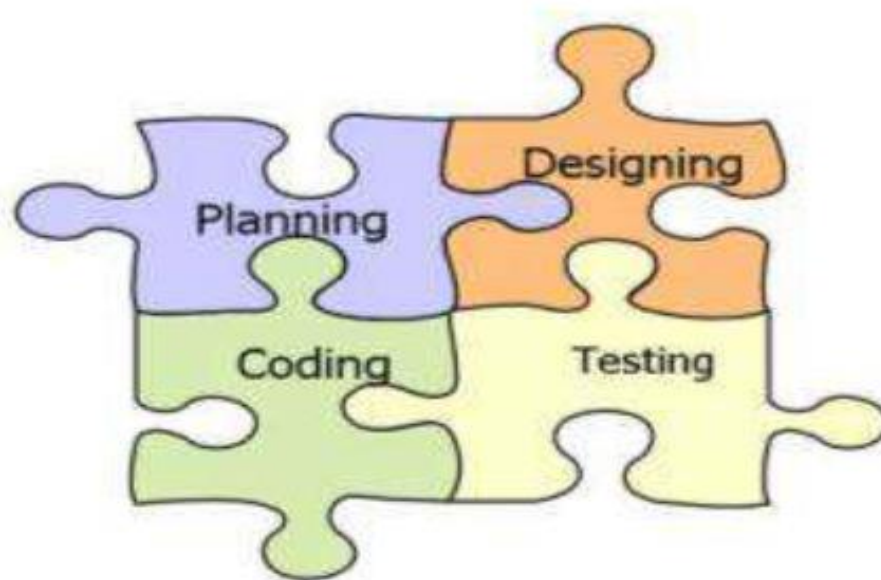


Figura 2. Actividades que guían el desarrollo de la metodología *XP*

La metodología *XP* es recomendable emplearla solo en proyectos a corto plazo e implica altas comisiones en caso de tener fallas en el desarrollo. Además tiene como desventaja que las tareas que se van entregando a los clientes son susceptibles a las modificaciones. El equipo de desarrollo sigue estrictamente el orden de prioridad de las tareas definidas por el cliente. Es una metodología demasiado costosa e innecesaria.

1.6.3 *Open Unified Process (OpenUp)*

OpenUp (*Open Unified Process* o Proceso Unificado Abierto) es un proceso modelo y extensible, dirigido a la gestión y desarrollo de proyectos de *software* basados en desarrollo iterativo, ágil e incremental. *OpenUp* es apropiado para proyectos pequeños de bajos recursos; y es aplicable a un conjunto amplio de plataformas y aplicaciones de desarrollo. Es parte del *framework* (infraestructura digital) de modelo de proceso de *Eclipse* (*Eclipse Process Framework*), desarrollado por la fundación *Eclipse*, que lo publica bajo una licencia libre [26].

Su ciclo de vida consta de 4 fases:

1. **Concepción:** Primera de las 4 fases en el ciclo de vida del proyecto, acerca del entendimiento del propósito y los objetivos, permite obtener suficiente información para confirmar que el proyecto debe hacer. El objetivo de esta fase es capturar las necesidades de los *stakeholder*⁸ en los objetivos del ciclo de vida del proyecto.
2. **Elaboración:** Es la segunda de las 4 fases del ciclo de vida de *OpenUP* donde se tratan los riesgos significativos para la arquitectura. El propósito de esta fase es establecer la base de la elaboración de la arquitectura del sistema.
3. **Construcción:** Esta fase está enfocada al diseño, implementación y prueba de las funcionalidades para desarrollar un sistema completo. El objetivo de esta fase es completar el desarrollo del sistema basado en la arquitectura definida.
4. **Transición:** Es la última fase, cuyo propósito es asegurar que el sistema es entregado a los usuarios, y evalúa la funcionalidad y rendimiento del último entregable de la fase de construcción.

Cada una de las fases se desarrolla mediante un ciclo de iteraciones como muestra la **Figura 3** [26].

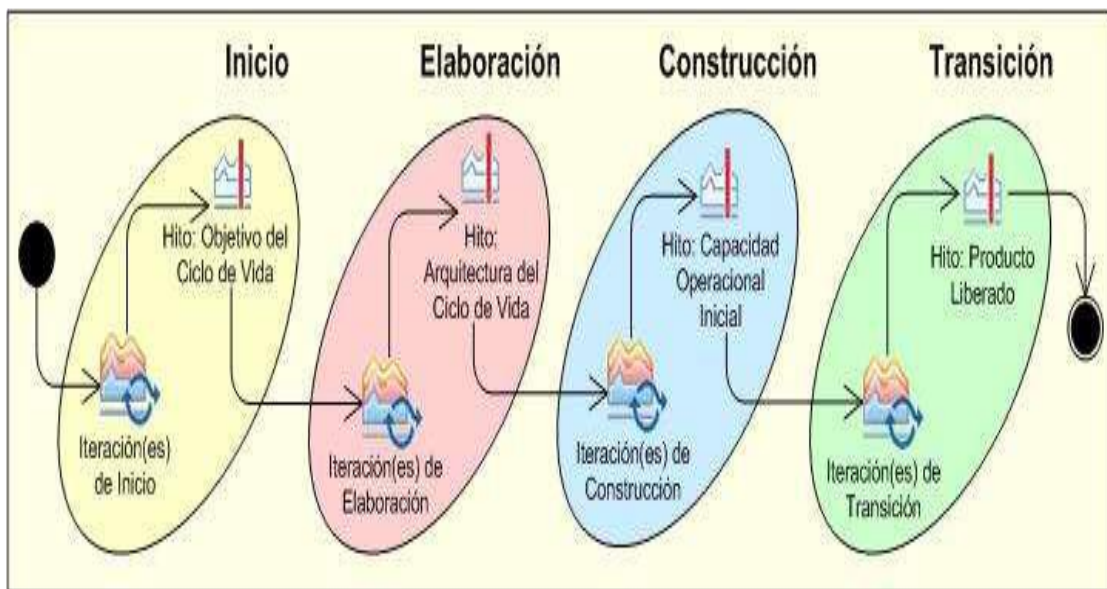


Figura 3. Fases del ciclo de vida de *OpenUp*

⁸ Persona que es significativamente afectada por el resultado del producto. Representa los intereses cuyas necesidades deben ser satisfechas por el proyecto.

1.6.4 Agile Unified Process (AUP)

AUP (*Agile Unified Process* o Proceso Unificado Ágil de *Scott Ambler*) es una versión simplificada de *RUP*. Este describe de una manera simple y fácil de entender la forma de desarrollar aplicaciones de *software* de negocio usando técnicas ágiles y conceptos que aún se mantienen válidos en *RUP*. *AUP* aplica técnicas ágiles incluyendo [27]:

- ✓ *TDD* (*Test Driven Development* o Desarrollo Dirigido por Pruebas)
- ✓ Modelado ágil
- ✓ Gestión de cambios ágil
- ✓ Refactorización de Base de Datos para mejorar la productividad.

Al igual que en *RUP*, en *AUP* se establecen cuatro fases que transcurren de manera consecutiva. Las cuales se muestran a continuación:

1. **Inicio:** El objetivo de esta fase es obtener una comprensión común cliente-equipo de desarrollo del alcance del nuevo sistema y definir una o varias arquitecturas candidatas para este.
2. **Elaboración:** El objetivo es que el equipo de desarrollo profundice en la comprensión de los requisitos del sistema y en validar la arquitectura.
3. **Construcción:** Durante la fase de construcción el sistema es desarrollado y probado al completo en el ambiente de desarrollo.
4. **Transición:** El sistema se lleva a los entornos de preproducción donde se somete a pruebas de validación y aceptación y finalmente se despliega en los sistemas de producción.

Cada una de las fases se desarrolla mediante un ciclo de iteraciones como muestra la **Figura 4** [27].

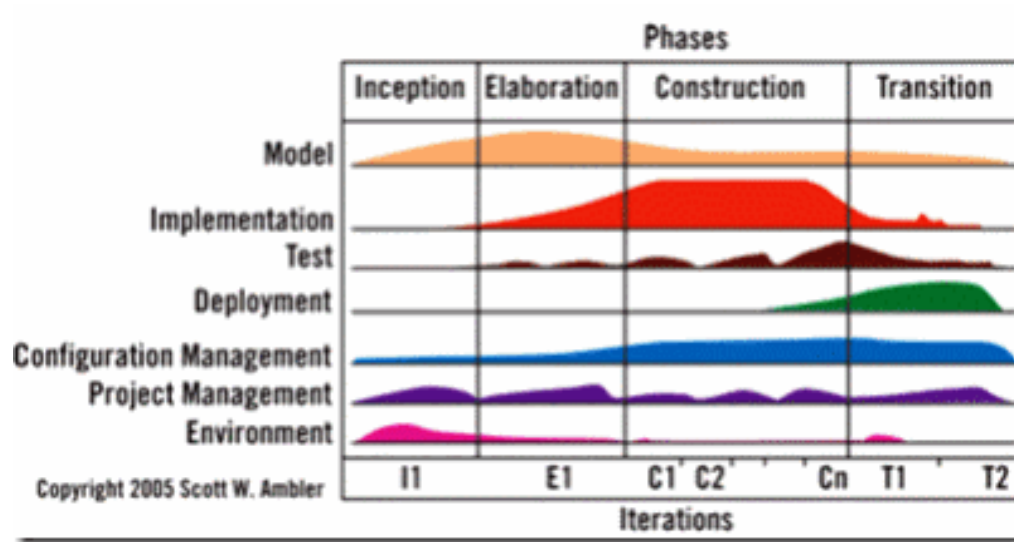


Figura 4. Fases del ciclo de vida de AUP

Al no existir una metodología de *software* universal, y que toda metodología puede ser adaptada a las características de cada proyecto. Se decide utilizar una variación de la metodología AUP apoyándose en el Modelo CMMI-DEV v1.3. De forma tal que se adapte al ciclo de vida definido para el desarrollo del portal web. CMMI, como sus siglas en inglés lo indican: *Capability Maturity Model Integration*. Básicamente es un modelo que se usa en los procesos para desarrollar productos y/o servicios [28]. Este es utilizado por las organizaciones para entender las mejores prácticas de la industria, priorizar y adoptar las mejoras a los procesos existentes, compararse con su competencia dentro del mercado para que los clientes puedan identificar las prácticas que necesitan demostrar sus proveedores. Su función es la mejora de procesos de desarrollo en las empresas.

De las 4 fases que propone AUP (Inicio, Elaboración, Construcción, Transición) se decide mantener la fase de Inicio, pero modificando el objetivo de esta, se unifican las restantes 3 fases de AUP en una sola, la que se llama ejecución y se agrega una fase de cierre.

Objetivos de las Fases de AUP (Para el proyecto):

1. **Inicio:** Durante el inicio del proyecto se llevan a cabo las actividades relacionadas con la planeación del proyecto. En esta fase se realiza un estudio inicial de la organización cliente que permite obtener información fundamental acerca del alcance del proyecto, se realiza estimación de tiempo, esfuerzo y costo y se decide si se ejecuta o no el proyecto.

2. **Ejecución:** En esta fase se ejecutan las actividades requeridas para desarrollar el *software*, incluyendo el ajuste de los planes del proyecto considerando los requisitos y la arquitectura. Durante el desarrollo se modela el negocio, obtienen los requisitos, se elaboran la arquitectura y el diseño, se implementa y se libera el producto. El producto es transferido al ambiente de los usuarios finales o entregado al cliente. Además, en la transición se capacita a los usuarios finales sobre la utilización del *software*.
3. **Cierre:** En esta fase se analizan tanto los resultados del proyecto como su ejecución y se realizan las actividades formales de cierre del proyecto.

Cada una de las fases se desarrolla mediante un ciclo de iteraciones como muestra la **Figura 5** [29].

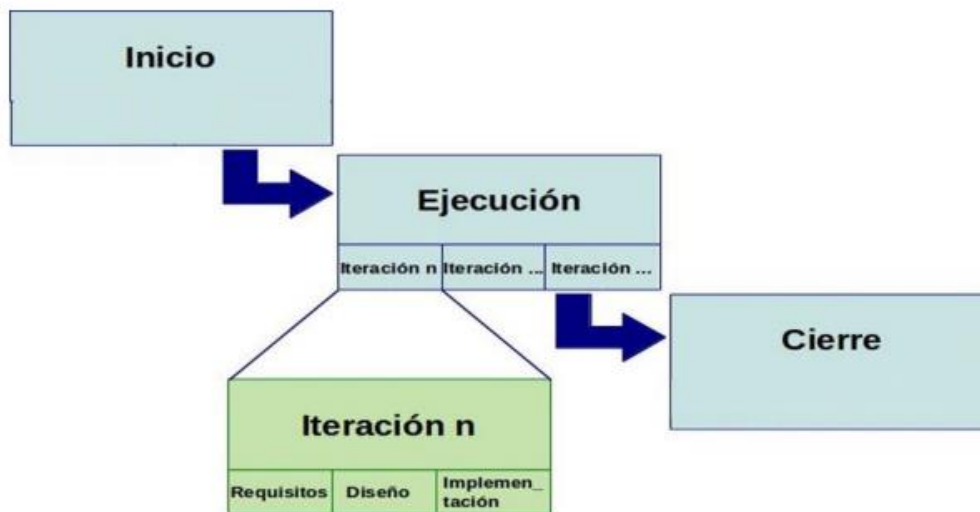


Figura 5. Fases del ciclo de vida de AUP para el proyecto

Estas modificaciones permiten un mayor éxito en la entrega del producto, ya que no se enfoca solamente en cubrir las necesidades del cliente. Esto se puede verificar al analizar lo integrado por cada fase, por ejemplo: en la fase inicial además de obtener una buena comprensión entre el cliente y el equipo de trabajo, se desarrolla una buena planificación, se estima el tiempo de desarrollo, el esfuerzo y el costo verificando si es factible desarrollar el producto en cuestión. En la fase de ejecución se realizan todas las actividades pertinentes a lo que originalmente se desarrollan en elaboración, construcción y transición, además que el despliegue que está comprendido en la fase de transición, originalmente, no es obligatorio ya que es decisión del equipo de trabajo poder capacitar al usuario final del producto en cuestión. Se agrega finalmente una fase de cierre donde como se describe anteriormente se analizan los resultados del proyecto.

Otra ventaja que proporciona es que entre las técnicas ágiles que utiliza *AUP* se encuentra el modelado ágil, se hace uso de esta técnica para los proyectos que necesiten por sus características encapsular sus requisitos funcionales en historias de usuarios (HU) o en Descripción de requisitos por procesos. La otra forma de encapsular los requisitos se mantiene por casos de uso (CU) [29].

1.6.5 Selección de la metodología de desarrollo de *software*

Debido al tamaño del equipo de trabajo y las necesidades de liberar el producto en un corto plazo de tiempo, es necesario guiar el proceso con una metodología ágil y que permita generar solamente aquellos artefactos que sean necesarios para el desarrollo del producto. Al realizar un estudio de diversas metodologías se considera no utilizar la metodología *RUP* debido a las características antes mencionadas. Luego se analizaron *OpenUp*, *XP* y *AUP* llegando a la conclusión que aunque las tres tengan características similares, se selecciona *AUP* ya que ofrece múltiples ventajas y es la que mejor satisface las necesidades del equipo de trabajo.

1.7 Conclusiones parciales

Después de realizar un estudio de diversas bibliografías, se arribó a las siguientes conclusiones:

- ✓ La caracterización de las tendencias actuales de los portales web dedicados a la divulgación de la información política, permitió identificar las principales funcionalidades que debe presentar la propuesta de solución.
- ✓ El análisis de las diferentes metodologías, tecnologías y herramientas permitió determinar las más adecuadas para la elaboración del portal web del PCC.

Capítulo 2 Características de la propuesta de solución

2.1 Introducción

En el presente capítulo se presentan las características y el funcionamiento del sistema, donde se tiene en cuenta los estilos arquitectónicos y patrones de diseño aplicables al *CMS Drupal*. Al mismo tiempo se muestra a través de un modelo de dominio los principales objetos del sistema. Se identifican los requisitos funcionales y no funcionales que deben estar presentes en la solución. Mediante el modelo de despliegue se muestra una idea de cómo queda la distribución física del sistema.

2.2 Características del sistema

A partir del estudio exhaustivo de los sistemas homólogos se evidenció que poseen características comunes en cuanto a secciones, servicios y contenidos. Esto hizo posible la elección de las principales características que se desean promover en el portal. Para dar cumplimiento al problema planteado se desarrollará un portal web utilizando el *CMS Drupal*, que permita controlar de forma dinámica la información.

El portal web contará con varias secciones, en las que se presentarán de forma organizada los contenidos. En algunos de estos contenidos los usuarios tendrán las opciones de evaluarlos, compartirlos en las redes sociales y comentarlos. Además será posible descargar documentos desde la biblioteca y estará habilitado un foro para la discusión y el intercambio de información entre los usuarios sobre diferentes temáticas de interés para el PCC.

Los usuarios podrán suscribirse al boletín para recibir por correo electrónico notificaciones relacionadas con las noticias, ver los contenidos en los idiomas definidos (español o inglés) según lo seleccione y participar en las encuestas que estén abiertas en el portal. También se permitirá efectuar la búsqueda de los contenidos publicados en las que el usuario puede entrar palabras clave. La información del portal web será socializada a través de las redes sociales y canales como *Facebook*, *Twitter* y *RSS*, y tendrá soporte para su acceso desde dispositivos móviles. Contará con el mapa del sitio para un mayor conocimiento de la estructura del portal web, página con información de contacto y enlace a sitios de interés.

El portal web permitirá efectuar de forma automatizada sus publicaciones. Con este nuevo mecanismo quedará conformada la siguiente jerarquía de roles, donde cada uno tendrá acceso específico sobre los contenidos del portal:

- ✓ **Usuario:** Usuarios que navegan en el portal web sin estar autenticados. Puede ver todos los contenidos que estén publicados en el portal, comentarlos y compartirlos en las redes sociales, y además pueden votar en las encuestas.
- ✓ **Bloguero:** Usuarios que escriben los *blogs* en el portal. Solamente tienen permisos para gestionar los *blog* publicados por él.
- ✓ **Periodista de provincia:** Usuarios que escriben las noticias en el portal. Solamente tienen el privilegio de gestionar las noticias publicadas por él.
- ✓ **Grupo editorial:** Son los responsables de gestionar y realizar las traducciones de los contenidos del portal.
- ✓ **Webmaster:** Es el responsable de crear y definir los permisos de acceso de los usuarios. Su principal tarea es mantener el portal actualizado y gestionar los servicios que va a brindar, además puede bloquear y desbloquear direcciones ip. Este usuario va a contar con todos los privilegios de los roles mencionados anteriormente.

2.3 Diagrama del modelo de dominio de la solución

El modelo de dominio es una representación visual de las clases conceptuales u objetos del mundo real en un dominio de interés [30]. Su uso posibilita ilustrar los elementos involucrados en el proyecto, así como las interrelaciones entre estos. A partir de su implementación se puede modelar el negocio asociado a la solución propuesta.

Para el desarrollo del portal web del PCC se decide representar este modelo ya que el proceso en dicho portal no es tan complejo como para desarrollar un modelo de negocio. La siguiente figura muestra el diagrama del modelo de dominio de la solución propuesta:

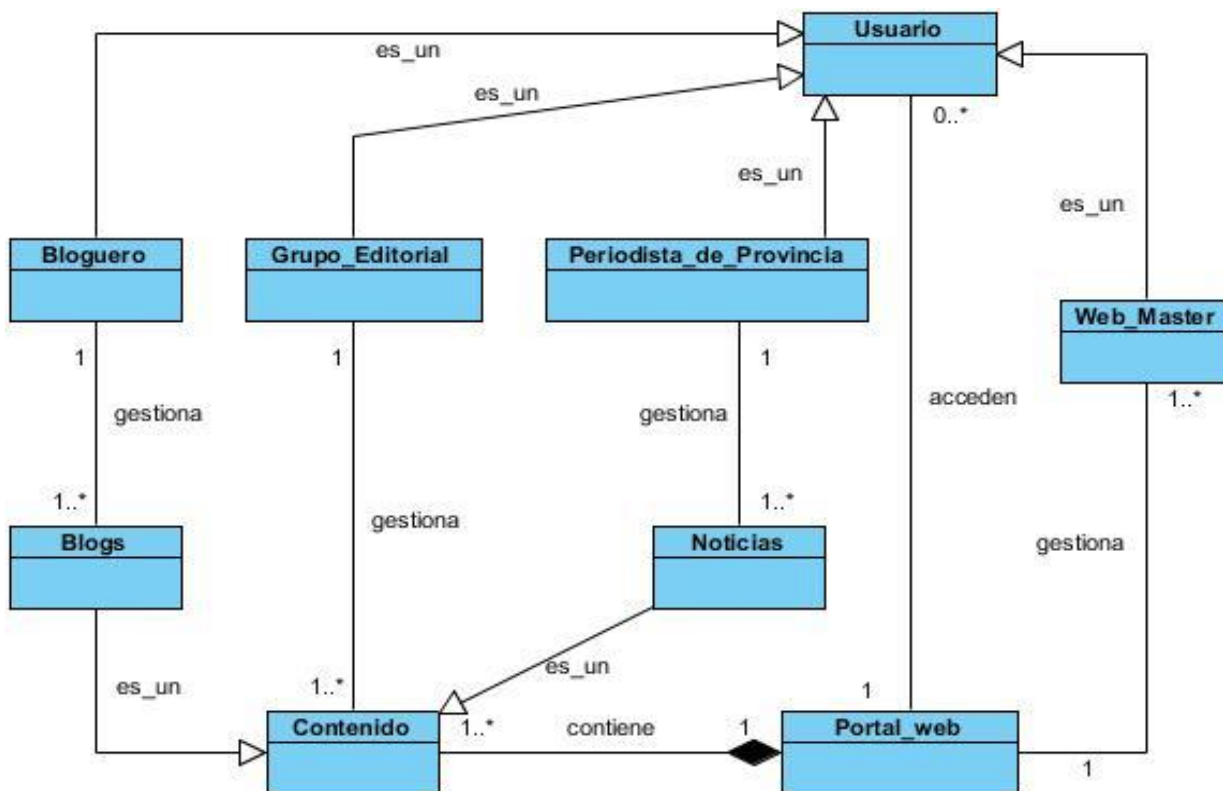


Figura 6. Diagrama de clases del modelo de dominio del sistema

A continuación se explican los conceptos asociados a las clases fundamentales del diagrama:

- ✓ **Portal web:** Contiene los contenidos y los servicios que brinda el portal.
- ✓ **Contenido:** Representa todos los contenidos que se gestionan en el portal.
- ✓ **Noticias:** Representa el contenido noticias, fue necesario representarlo así para especificar en el diagrama que los usuarios con rol Periodista de provincia solo gestiona noticias.
- ✓ **Blogs:** Representa el contenido *blog*, fue necesario representarlo así para especificar en el diagrama que el usuario Bloguero solo gestiona *blog*.

2.4 Especificaciones de los requisitos del software

La especificación de los requisitos del *software* se encarga de definir y describir de forma clara, consistente, compacta, y sin ambigüedades, el comportamiento del sistema. Disminuye los retrasos del desarrollo del proyecto y permite estimar costos, tiempo y recursos. Se dividen en dos grupos: los requisitos funcionales

que son las condiciones que un producto debe cumplir y los no funcionales que representan las cualidades que el producto debe tener.

2.4.1 Requisitos Funcionales

RF1 Autenticar usuario	RF2 Crear usuario
RF3 Editar usuario	RF4 Eliminar usuario
RF5 Mostrar usuario	RF6 Crear noticia
RF7 Editar noticia	RF8 Eliminar noticia
RF9 Mostrar noticia	RF10 Mostrar noticia 5 Héroes
RF11 Crear Tema del foro	RF12 Editar Tema del foro
RF13 Eliminar Tema del foro	RF14 Mostrar Tema del foro
RF15 Comentar Tema del foro	RF16 Crear Galería de videos
RF17 Editar Galería de videos	RF18 Eliminar Galería de videos
RF19 Mostrar Galería de videos	RF20 Crear Galería de imágenes
RF21 Editar Galería de imágenes	RF22 Eliminar Galería de imágenes
RF23 Mostrar Galería de imágenes	RF24 Crear Galería de infografías
RF25 Editar Galería de infografías	RF26 Eliminar Galería de infografías
RF27 Mostrar Galería de infografías	RF28 Crear Galería de audio
RF29 Editar Galería de audio	RF30 Eliminar Galería de audio
RF31 Mostrar Galería de audio	RF32 Crear Galería de multimedia
RF33 Editar Galería de multimedia	RF34 Eliminar Galería de multimedia
RF35 Mostrar Galería de multimedia	RF36 Búsqueda simple
RF37 Búsqueda avanzada	RF38 Mostrar Página básica
RF39 Editar Página básica	RF40 Crear Suceso Histórico
RF41 Editar Suceso Histórico	RF42 Eliminar Suceso Histórico
RF43 Mostrar Suceso Histórico	RF44 Crear Personalidades
RF45 Editar Personalidades	RF46 Eliminar Personalidades
RF47 Mostrar Personalidades	RF48 Crear Documento Rector
RF49 Editar Documento Rector	RF50 Eliminar Documento Rector
RF51 Mostrar Documento Rector	RF52 Crear Evento
RF53 Editar Evento	RF54 Eliminar Evento
RF55 Mostrar Evento	RF56 Crear Miembro de estructura
RF57 Editar Miembro de estructura	RF58 Eliminar Miembro de estructura
RF59 Mostrar Miembro de estructura	RF60 Realizar Comentario
RF61 Aprobar Comentario	RF62 Eliminar Comentario
RF63 Crear Congreso	RF64 Editar Congreso
RF65 Eliminar Congreso	RF66 Mostrar Congreso
RF67 Enviar Atención a la población administradores	RF68 Enviar Contactar con
RF69 Crear Contenido relevante	RF70 Editar Contenido relevante
RF71 Eliminar Contenido relevante	RF72 Mostrar Contenido relevante
	RF74 Editar Relación internacional

<p>RF73 Crear Relación internacional</p> <p>RF75 Eliminar Relación internacional</p> <p>RF77 Crear Delegación</p> <p>RF79 Eliminar Delegación</p> <p>RF81 Crear Documento</p> <p>RF83 Eliminar Documento</p> <p>RF85 Crear Proceso legal</p> <p>RF87 Eliminar Proceso legal</p> <p>RF89 Crear Video portada</p> <p>RF91 Eliminar Video portada</p> <p>RF93 Crear Encuesta</p> <p>RF95 Eliminar Encuesta</p> <p>RF97 Mostrar Resultados de la encuesta</p> <p>RF99 Mostrar Selector de idioma</p> <p>RF101 Imprimir</p> <p>RF103 Enviar por Correo</p> <p>RF105 Editar Enlaces de interés</p> <p>RF107 Mostrar Enlaces de interés</p> <p>RF109 Mostrar RSS</p> <p>RF111 Editar Elemento del directorio</p> <p>RF113 Mostrar Elemento del directorio</p> <p>RF115 Editar Pregunta frecuente</p> <p>RF117 Mostrar Pregunta frecuente</p> <p>RF119 Editar Blog</p> <p>RF121 Mostrar Blog</p> <p>RF123 Suscribirse usuarios al boletín</p> <p>RF125 Crear boletín</p> <p>RF127 Eliminar boletín</p> <p>RF129 Mostrar Reflexiones de Fidel</p> <p>RF131 Crear Anuncio</p> <p>RF133 Eliminar Anuncio</p> <p>RF135 Bloqueo de IP</p> <p>RF137 Mostrar IP bloqueados</p> <p>RF139 Editar Traducción de contenido</p> <p>RF141 Crear Termino</p> <p>RF143 Eliminar Termino</p>	<p>RF76 Mostrar Relación internacional</p> <p>RF78 Editar Delegación</p> <p>RF80 Mostrar Delegación</p> <p>RF82 Editar Documento</p> <p>RF84 Mostrar Documento</p> <p>RF86 Editar Proceso legal</p> <p>RF88 Mostrar Proceso legal</p> <p>RF90 Editar Video portada</p> <p>RF92 Mostrar Video portada</p> <p>RF94 Editar Encuesta</p> <p>RF96 Mostrar Encuesta</p> <p>RF98 Mostrar Mapa del sitio</p> <p>RF100 Descargar en formato pdf</p> <p>RF102 Registrar votos de un contenido en la web</p> <p>RF104 Crear Enlaces de interés</p> <p>RF106 Eliminar Enlaces de interés</p> <p>RF108 Mostrar Enlaces a Redes Sociales</p> <p>RF110 Crear Elemento del directorio</p> <p>RF112 Eliminar Elemento del directorio</p> <p>RF114 Crear Pregunta frecuente</p> <p>RF116 Eliminar Pregunta frecuente</p> <p>RF118 Crear Blog</p> <p>RF120 Eliminar Blog</p> <p>RF122 Compartir</p> <p>RF124 Dar baja a usuarios del boletín</p> <p>RF126 Editar boletín</p> <p>RF128 Mostrar boletín</p> <p>RF130 Mostrar Discursos de Raúl</p> <p>RF132 Editar Anuncio</p> <p>RF134 Mostrar Anuncio</p> <p>RF136 Desbloquear IP</p> <p>RF138 Crear Traducción de contenido</p> <p>RF140 Eliminar Traducción de contenido</p> <p>RF142 Editar Termino</p> <p>RF144 Mostrar Termino</p>
--	---

Tabla 1. Requisitos Funcionales

2.4.2 Descripción de requisitos funcionales

En este acápite se muestran ejemplos de las descripciones de los requisitos funcionales de mayor impacto en el portal web, dicha descripción se realiza mediante historias de usuario (HU). Se exhiben más ejemplos en el [Anexo 1](#). Las descripciones de los restantes requisitos se pueden observar en el documento descripción de requisito ágil del portal web del PCC.

Número: RF9		Nombre del requisito: Mostrar Noticias	
Programador: Leonardo Gómez Romay		Iteración Asignada: 1	
Prioridad: Baja		Tiempo Estimado: 3 días	
Riesgo en Desarrollo: Ninguno		Tiempo Real: 3 días	
<p>Descripción: Una noticia se puede mostrar por la vista de Usuario y por la vista de Administración.</p> <p>Vista Usuario El usuario autenticado accede a la página principal del portal donde se mostrarán hasta 3 noticias relevantes. Si el usuario desea acceder a la noticia dará clic encima del título de esta y podrá visualizar toda la información de la noticia. Si el usuario desea ver el resto de las noticias deberá seleccionar la opción: “Ver más noticias”. Si el usuario desea acceder a la noticia de manera ampliada dará clic encima de esta, allí visualizará un listado de noticias. Las noticias se organizan por orden cronológico de forma ascendente. Si el usuario desea ver el resto de una noticia deberá dar clic encima del título de esta y podrá visualizar toda la información de la noticia.</p> <p>Vista Administración El usuario autenticado en el sistema con privilegios de administración accede al módulo “Contenido”, ubicado en el panel superior de la Intranet. El sistema le muestra al usuario un listado de todos los contenidos existentes en el sistema, seguido de las opciones: “editar” y “eliminar”. El usuario selecciona en el formulario “Mostrar solo los elementos en los que” los contenidos en los que el tipo sea noticia. El sistema le muestra al usuario un listado con las noticias existentes en el sistema, finalizando así la HU.</p>			
<p>Observaciones: 1.-Si no existen noticias en el sistema, no mostrará ninguna información.</p>			
Prototipo de interfaz:			

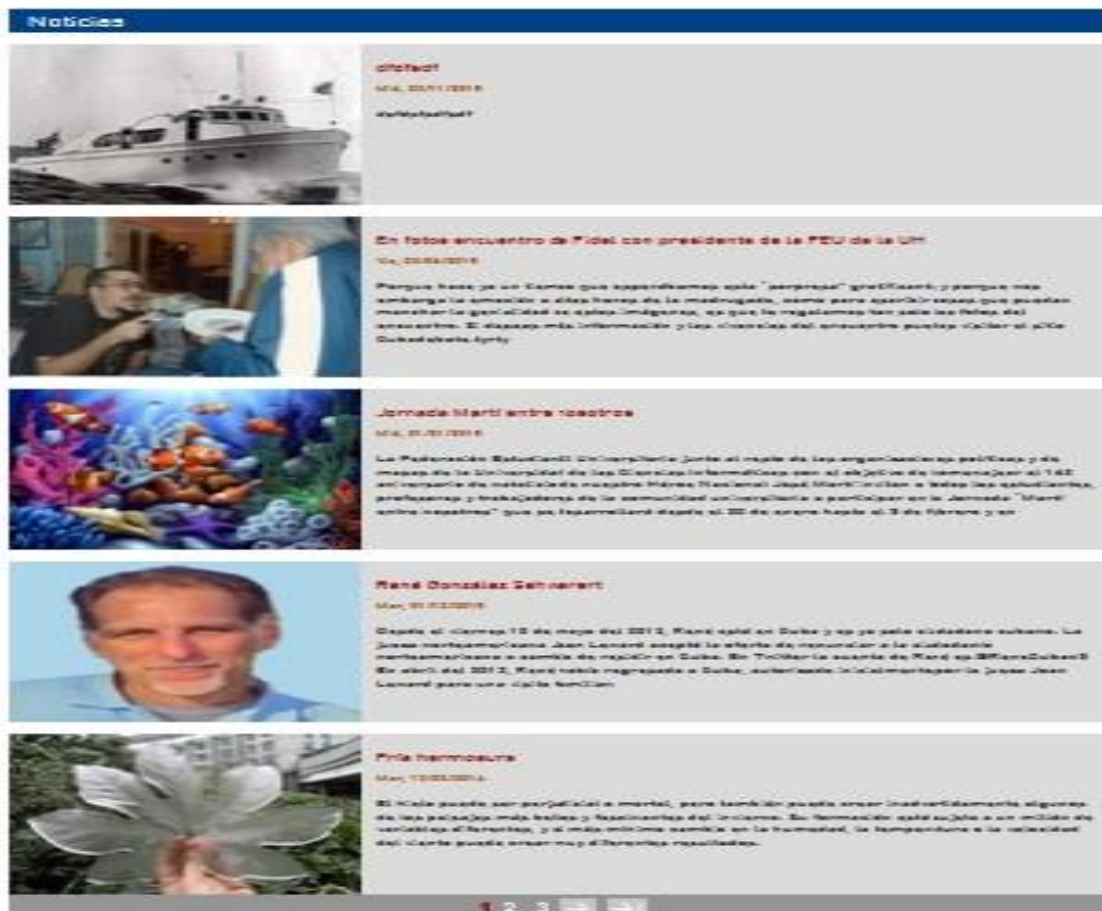


Tabla 2. Historia de usuario RF9 Mostrar Noticias

2.4.3 Requisitos no funcionales

Usabilidad

RNF1 El sistema podrá ser usado sobre ambiente web por personas con pocos conocimientos de informática.

RNF2 El sistema mostrará un mapa del sitio para garantizarle al usuario facilidad de navegación dentro del sistema.

Confiabilidad

RNF3 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 sistema, solo podrá mostrarse detalles de la información para los usuarios con privilegios de administración dentro del sistema.

RNF4 El sistema establecerá mecanismos para garantizar la confiabilidad e integridad de la información ante posibles accesos no autorizados.

RNF5 Se podrá acceder a la aplicación desde cualquier navegador web, ya sea en Internet *Explorer*, *Mozilla Firefox*, *Chrome*, *Opera* y *Safari* de sus versiones actuales.

Eficiencia

RNF6 Serán optimizados las imágenes, el diseño y el código fuente de la aplicación para garantizar un alto rendimiento del portal web.

RNF7 El sistema debe ser capaz de responder con rapidez a las peticiones de los usuarios, demorando como promedio en una transición, de uno 1 a cinco 5 segundos aproximadamente.

Soporte

RNF8 El sistema debe dar la posibilidad de ser mejorado, así como de incorporarle nuevas funcionalidades en caso de ser necesarias.

Interfaz

RNF9 El diseño cumple con los estándares internacionales de desarrollo web, ya que cuenta con una interfaz sencilla, intuitiva, que se encuentra acorde garantizando mayor nivel de usabilidad.

Interfaz de *Hardware* (Recomendado)

RNF10 PC Servidor de BD

Capacidad de almacenamiento en HDD: 320 GB

Capacidad de memoria RAM: 4 GB

Procesador: Core 2Duo 2.20Hz

RNF11 PC Servidor web

Capacidad de almacenamiento en HDD: 320 GB

Capacidad de memoria RAM: 4 GB

Procesador: Core 2Duo 2.20Hz

Interfaz de software**RNF12** PC Servidor web

Sistema operativo: Ubuntu 14.10 o superior

Servidor web: *Apache 2.2*

Lenguaje de programación: *PHP5* o superior

RNF13 PC Servidor de BD

Sistema Operativo: Ubuntu 14.10 o superior

Sistema Gestor de Base de Datos *PostgreSQL v9.1*

Requisitos legales, de derecho de autor y otros

RNF14 Las tecnologías y herramientas utilizadas están bajo licencia *GPL*.

RNF15 El sistema posee una sección donde se publican los términos y condiciones de uso que reglamentan el acceso y uso del portal web y de todos sus subdominios o dominios secundarios incluyendo sus contenidos y servicios, puestos a disposición del portal web que deben ser de estricto cumplimiento por parte de los usuarios.

RNF16 Los derechos de autor sobre la información de terceros que se publica en el portal web pertenecen a sus respectivos autores y su publicación en dicho portal se realiza sin ánimos de lucro y con un propósito informativo.

RNF17 El portal web del PCC contendrá vínculos a sitios web que son controlados y mantenidos por terceros, pero no se hace responsable por el contenido ni por la disponibilidad de esa información.

2.5 Arquitectura y patrones de diseño**2.5.1** Arquitectura de *Drupal*

Al utilizar el *CMS Drupal* para el desarrollo del portal web, la arquitectura y los patrones son heredados de este. En lugar de considerar el sitio web como un conjunto de páginas interrelacionadas, *Drupal* estructura los contenidos en una serie de elementos básicos. Estos son los datos (*data*), módulos (*modules*), bloques y menús (*blocks and menus*), permisos de usuario (*user permissions*) y plantillas (*templates*) [31].

Seguidamente se muestra una imagen que representa esta arquitectura y además se explica cada una de estas capas mencionadas anteriormente.

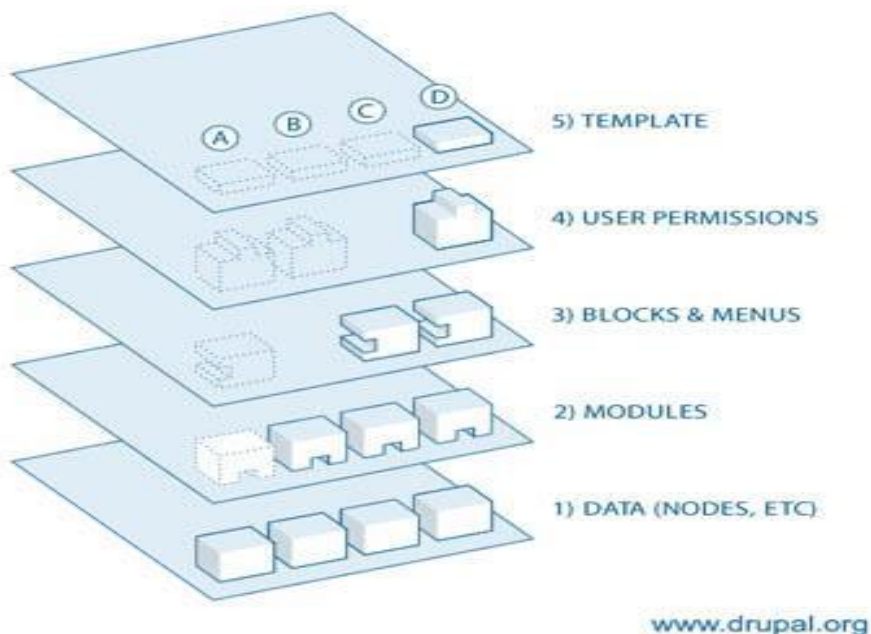


Figura 7. Estructura de la aplicación

- ✓ **Datos:** son los elementos básicos en que *Drupal* almacena la información, los contenidos. Así a medida que el portal web crece, lo va haciendo el número de nodos los cuales van formando un “depósito de nodos” cada vez mayor.
- ✓ **Módulos:** Son los elementos que operan sobre los nodos y otorgan funcionalidad a *Drupal* permitiendo incrementar sus capacidades o adaptarlas a las necesidades de cada portal web. Son complementos que se instalan en el portal web proporcionándole nuevas funcionalidades.
- ✓ **Bloques y Menú:** Estos permiten estructurar y organizar los contenidos en el portal web.
- ✓ **Permisos de usuario:** La mayoría de los portales web son multiusuario, por lo que la seguridad y control de los usuarios es un punto clave para garantizar la integridad de la información almacenada. Con esta finalidad *Drupal* dispone de un registro de usuarios y de roles que permiten especificar que tareas pueden realizar y a que contenidos puede acceder cada tipo de usuario.
- ✓ **Plantillas:** Es la que establece la apariencia gráfica o estilo de la información que se le muestra al usuario. Esta separación entre información y aspecto gráfico permite cambiar el diseño u apariencia del portal web sin necesidad de modificar los contenidos.

2.5.2 Patrones de diseño en *Drupal*

Un patrón de diseño provee un esquema para refinar los componentes de un sistema de *software*, o las relaciones entre ellos. Estos brindan soluciones a los problemas que pueda haber en el diseño de un *software*.

Drupal utiliza patrones de diseños propiamente orientados a objetos como los patrones *Gang of Four (GoF)*, permitiendo la construcción de un diseño elegante y robusto. Los patrones *GoF* son clasificados según el propósito para el cual han sido definidos [32]. Entre los patrones usados por *Drupal* se encuentran los siguientes [33]:

- **Patrones de creación:** solucionan problemas de creación de instancias. No ayudan a encapsular y abstraer dicha creación.
 - ✓ **Instancia única (*Singleton*):** Garantiza que exista una única instancia de una clase proporciona un punto de acceso global a ella.

Si los módulos y los temas en *Drupal* se piensan como objetos, este patrón está presente, debido a que en general estos objetos no encapsulan datos, lo que separa a un módulo de otro es el conjunto de funciones que contiene. Esto garantiza la presencia de una única instancia y el mecanismo de acceso global a ella.
- **Patrones estructurales:** solucionan problemas de composición (agregación) de clases y objetos.
 - ✓ **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 bases de datos de *Drupal* se aplica de una forma similar al patrón de diseño bridge. Los módulos necesitan ser escritos de acuerdo a la *API* definida y de forma independiente del sistema que se utiliza en la base de datos, para que proporcionen la capa de abstracción para ello, añadiendo soporte para más sistemas de bases de datos sin la necesidad de modificar el código del módulo.
 - ✓ **Decorador (*Decorator*):** Asigna responsabilidades adicionales a un objeto dinámicamente, proporcionando una alternativa flexible a la herencia para extender la funcionalidad. Permite no tener que crear clases que hereden de la primera, incorporando nuevas funcionalidades, sino otras que la implementan y se asocian a ella.

En *Drupal* el uso de `hook_nodeapi()`, permite a los módulos extender arbitrariamente el comportamiento de todos los nodos. Para los archivos cargados y adjuntados a un nodo, se podría diseñar un nuevo tipo de nodo con las demás características del nodo, que brinde la posibilidad de cargar archivos con la concesión de que cada nodo solicite la posibilidad de tener archivos adjuntos. Este comportamiento podría ser imitado por el uso del patrón decorador envolviéndolo alrededor de cada objeto nodo.

➤ **Patrones de comportamiento:** solucionan problemas respecto a la interacción y responsabilidades entre clases y objetos, así como los algoritmos que encapsulan.

- ✓ **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 se realiza una modificación a un vocabulario en el sistema de taxonomía de *Drupal*, el *hook taxonomy* es llamado en todos los módulos que lo implementan. Por la aplicación del *hook*, estos módulos quedan registrados como observadores del objeto vocabulario; cualquier cambio que sufra, los módulos dependientes del objeto actúan tanto como sea apropiado.

- ✓ **Orden (*Command*):** Encapsula una petición en un objeto, permitiendo así parametrizar a los clientes con diferentes peticiones, hacer cola o llevar un registro de las peticiones y poder deshacer las operaciones.

Muchos de los *hooks* (ganchos) de *Drupal* utilizan el patrón de mando para reducir el número de funciones que son necesarias para la aplicación, pasando la operación como un parámetro, junto con los argumentos. El propio sistema de *hook* utiliza este patrón, a fin de que los módulos no tengan que definir cada gancho.

- ✓ **Cadena de responsabilidades (*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 petición de una página, el sistema de menús de *Drupal* determina si hay algún módulo para responder la

petición, si el usuario tiene acceso al recurso solicitado y qué función se debe llamar para procesar la petición. En este proceso se transmite el mensaje de la petición por cada uno de los componentes que se encuentran inmersos. De esta forma se continúa la cadena hasta que un módulo atienda la petición, un módulo deniegue el acceso o la cadena se agote.

- ✓ **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.

2.6 Diagrama de despliegue del sistema

Se muestra a continuación el diagrama de despliegue correspondiente al sistema, donde se puede observar que se encontrará desplegado en una unidad de procesamiento, contenida dentro del Servidor de aplicaciones web (*Apache*). La base de datos estará desplegada en una unidad de procesamiento corriendo el servicio de *PostgreSQL 9.1*. La asociación entre los nodos representa la ruta de comunicación entre ellos.

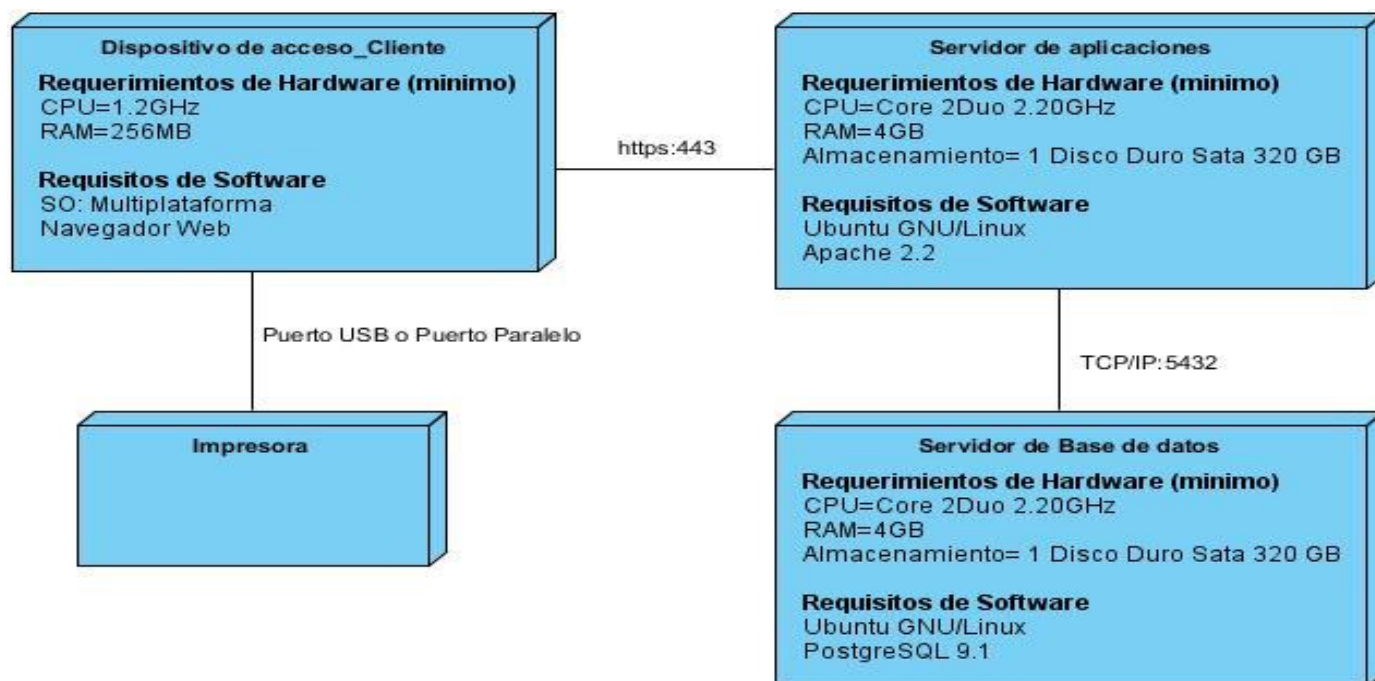


Figura 8. Diagrama de despliegue del sistema

Descripción de elementos e interfaces de comunicación:

<<**HTTPS**>>: Protocolo para establecer a través del puerto 443 la conexión segura entre el dispositivo de acceso cliente y el servidor de aplicaciones. La conexión es por cable vía *modem*, *LAN* o red inalámbrica con una velocidad de más de 64 Kbps.

<<**TCP/IP**>>: Estos protocolos establecen la conexión entre el servidor de aplicaciones y el servidor de base de datos. Para el servidor de base de datos de *PostgreSQL* se define el puerto 5432. La conexión entre el servidor web y el servidor de base de datos permite dar órdenes y obtener información de esta.

2.7 Conclusiones parciales

En el presente capítulo se realizó una descripción de las características con las que tiene que contar el portal y se describieron los diferentes requisitos funcionales y no funcionales que tiene que cumplir este para lograr una buena calidad. A través de la realización de todas estas actividades se arriba a la siguiente conclusión:

- ✓ Los requerimientos funcionales y no funcionales obtenidos a partir del proceso de identificación de los requisitos, sirven de guía para desarrollar las distintas funcionalidades de la propuesta de solución.
- ✓ Los artefactos generados constituyeron una guía fundamental para el desarrollo exitoso de la propuesta de solución.
- ✓ El diagrama de despliegue propuesto permitió exponer cómo se relacionan los elementos de *hardware* del sistema.

Capítulo 3 Implementación y pruebas de la propuesta de solución

3.1 Introducción

En el siguiente capítulo se presenta el diagrama de componentes y la definición de los estándares de codificación que tienen en cuenta los desarrolladores en la implementación del portal web. Además se muestra una descripción de las pruebas seleccionadas para validar la implementación realizada, y se brindan los resultados obtenidos en la ejecución de cada una de ellas.

3.2 Diagrama de componentes

El diagrama de componentes es un gráfico que muestra los componentes de un sistema de *software* conectados por las relaciones de dependencias lógicas entre cada uno de ellos. Provee una vista arquitectónica de alto nivel del sistema, ayudando a los desarrolladores a visualizar el camino de la implementación. La realización del diagrama posibilita tomar decisiones respecto a las tareas de implementación y los requisitos [34].

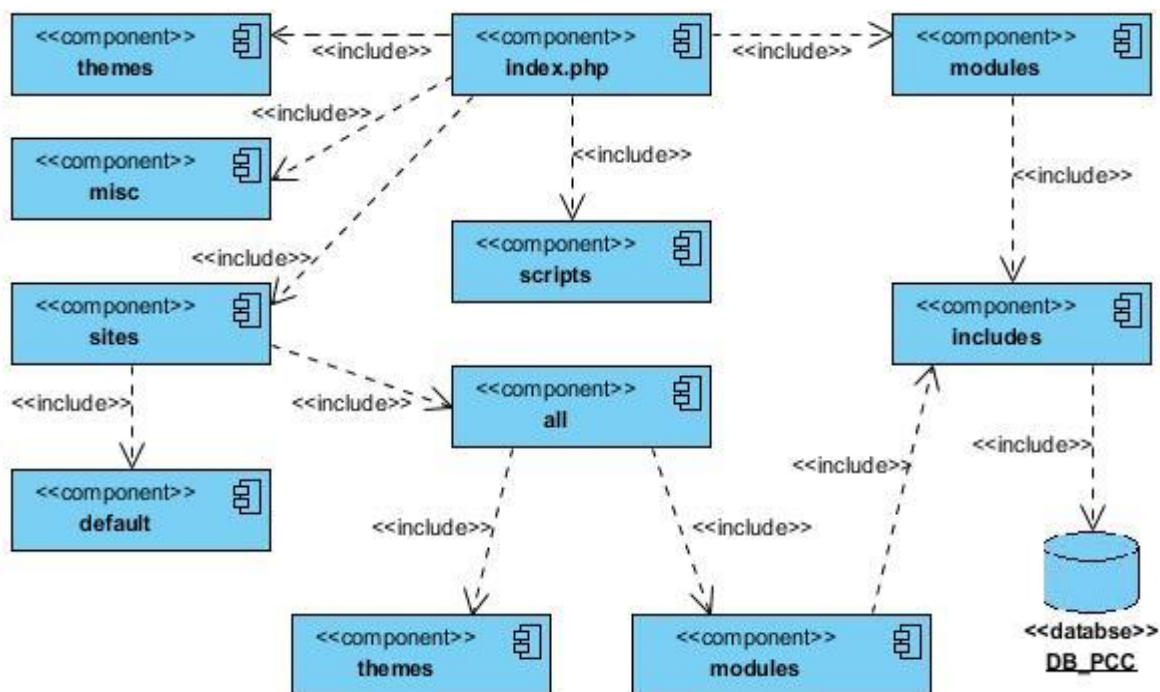


Figura 9. Diagrama de componentes del sistema

A continuación se describen cada uno de los componentes del diagrama:

index.php: El archivo `index.php` es la puerta de entrada al sistema. A partir de esta entrada se solicitan los diferentes módulos del *CMS Drupal*. Cuando cargamos una página del sitio web estamos siempre haciendo una llamada a `index.php`. El procesamiento de la página comienza en el *script* `index.php`, que llama a la función `drupal_bootstrap ()` y pone en marcha distintas fases del proceso de arranque del sistema (*bootstrap*).

Themes: Contiene los temas que vienen con la distribución de *Drupal*.

Scripts: Contiene utilidades adicionales que no utiliza *Drupal* directamente, pero que podemos utilizar desde la línea de comandos de *shell*. Por ejemplo, el *script* `password-hash.sh` permite obtener una contraseña codificada a partir de la contraseña original (en texto plano).

Misc: Este componente incluye archivos *JavaScript* e imágenes requeridas por el sistema.

Includes: Contiene un conjunto de librerías en forma de archivos *PHP* con extensión `.inc`, que incluyen funciones comunes del sistema indispensables para su funcionamiento, como son las conexiones a la base de datos.

Modules: Contiene todos los módulos del núcleo que permiten las distintas funcionalidades del *CMS*, cada uno en su carpeta correspondiente. Nunca se deben modificar directamente los módulos, ni subir módulos adicionales a esta carpeta.

Sites: La carpeta *sites* contiene los extras y modificaciones que se añaden a la distribución original. En esta se encuentran los módulos adicionales creados, añadidos o descargados del repositorio de módulos de *Drupal*, colocados en `sites/all/modules`. Además los temas adicionales instalados o creados, en `sites/all/themes`. La carpeta incluye tras la instalación el archivo de configuración del sitio (`settings.php`) [35].

3.3 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 se utilizan en la implementación de los módulos.

Para la declaración del código se tienen en cuenta los siguientes estándares [36]:

3.3.1 Indentación

La indentación consiste en insertar espacios en blanco o tabuladores en determinadas líneas de código para facilitar su comprensión. En programación se utiliza la indentación para anidar elementos.

En *Drupal* debemos indentar con 2 espacios, nunca con tabuladores. Además, no se debe dejar espacios en blanco al final de cada línea. En el siguiente ejemplo se muestra un fragmento de código con las indentaciones realizadas, de 2 espacios cada una, y los saltos de línea o enter al final de cada línea (sin dejar espacios).

```
<?php
function forum_block_view($delta = '') {
    $query = db_select('forum_index', 'f')
    $query->fields('f')
    $query->addTag('node_access');
    switch ($delta) {
        case 'active':
            $title = t('Active forum topics');
            $query
                ->orderBy('f.last_comment_timestamp', 'DESC')
                ->range(0, variable_get('forum_block_num_active', '5'));
            break;
        case 'new':
            $title = t('New forum topics');
            $query
                ->orderBy('f.created', 'DESC')
                ->range(0, variable_get('forum_block_num_new', '5'));
            break;
    }
    $block['subject'] = $title;
    // Cache based on the altered query.
    $block['content'] = drupal_render_cache_by_query($query,
        'forum_block_view');
    $block['content']['#access'] = user_access('access content');
    return $block;
}
?>
```

Figura 10. Identación del código

3.3.2 Etiquetas de apertura y cierre de *PHP*

Cuando se escribe en *PHP*, siempre se deben utilizar las etiquetas `<?php` y `?>`, y en ningún caso la versión corta `<? y ?>`. En general se omite la etiqueta de cierre de *PHP* (`?>`) al final de los archivos `.module` y `.inc`. Esta convención evita que se puedan quedar olvidados espacios no deseados al final del archivo (después de la etiqueta de cierre `?>`), que serían identificados como salida *HTML* y podrían provocar un error muy típico, "*Cannot modify header information - headers already sent by...*".

Por tanto, la etiqueta de cierre final del archivo (`?>`) es opcional en Drupal.

No hay que confundir esto con el uso normal del lenguaje *PHP* en archivos que también contienen *HTML* (como por ejemplo los archivos de plantilla `.tpl.php`), donde **cada fragmento de *PHP* debe llevar sus correspondientes etiquetas de apertura y cierre**, para diferenciarlo del código *HTML*.

En el siguiente ejemplo vemos cómo se utilizan las etiquetas de apertura y cierre de *PHP* en los archivos de plantilla `.tpl.php`. Si el último contenido del archivo es un fragmento de código *PHP*, deberá llevar su correspondiente etiqueta de cierre.

```

<?php

/**
 * @file
 * Default theme implementation to display a forum which may contain forum
 * containers as well as forum topics.
 *
 * Variables available:
 * - $forums: The forums to display (as processed by forum-list.tpl.php)
 * - $topics: The topics to display (as processed by forum-topic-list.tpl.php)
 * - $forums_defined: A flag to indicate that the forums are configured.
 *
 * @see template_preprocess_forums()
 * @see theme_forums()
 */
?>
<?php if ($forums_defined): ?>
<div id="forum">
  <?php print $forums; ?>
  <?php print $topics; ?>
</div>
<?php endif; ?>

```

Figura 11. Ejemplo de etiquetas de apertura y cierre *PHP*

3.3.3 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, `$numero = 3`, en lugar de `$numero=3`. Esto se aplica a operadores como `+`, `-`, `*`, `/`, `=`, `==`, `!=`, `>`, `<`, `.` (Concatenación de cadenas), `.=`, `+=`, `-=`, etc.

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

3.3.4 Uso de comillas

Se pueden usar tanto las comillas simples ('cadena') como las comillas dobles ("cadena") para delimitar las cadenas de caracteres.

Las comillas dobles son necesarias si se desean incluir variables dentro de las cadenas de texto. Por ejemplo, `"<h1>${title}</h1>"`. También se recomienda el uso de comillas dobles cuando el texto puede incluir alguna comilla simple.

3.3.5 Uso de punto y coma (;) en código PHP

Aunque PHP permite escribir líneas de código individuales sin el terminador de línea (;), como por ejemplo `<?php print $title ?>`. En Drupal es **siempre** obligatorio: `<?php print $title; ?>`.

- Correcto: `<?php print $title; ?>`

- Incorrecto: `<?php print $title ?>`

3.3.6 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 sitúa en la misma línea que la definición de la estructura, separada por un espacio.
- Se recomienda utilizar 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).

- Las estructuras *else* y *elseif* se escribirán en la línea siguiente al cierre de la sentencia anterior.

3.3.7 Funciones

Los nombres de las funciones deben estar escritos en minúsculas y las palabras separadas por guión bajo. Además, se debe incluir siempre como prefijo el nombre del módulo o tema, para evitar así duplicidad de funciones.

En su declaración, después del nombre de la función, el paréntesis de inicio de los argumentos debe ir sin espacio. Cada argumento debe ir separado por un espacio, después de la coma del argumento anterior.

Ejemplo:

```
function forum_help($path, $arg) {
```

En la llamada a la función se aplican las mismas reglas anteriores con respecto a los parámetros, como se muestra en el siguiente ejemplo:

```
$field = field_info_instance('node', 'taxonomy_forums', $node->type);
```

Como excepción, es posible usar más de un espacio antes de una asignación (=) para mejorar la presentación, cuando se estén realizando varias asignaciones en bloque:

```
$numero1 = foo($a, $type);
```

```
$primer_valor = foo2($b);
```

```
$i = foo3();
```

3.3.8 Arrays

Los valores dentro de un *array* (o matriz) se deben separar por un espacio (después de la coma que los separa). El operador => debe separarse por un espacio a ambos lados.

Cuando la línea de declaración del *array* supera los 80 caracteres, cada elemento se debe escribir en una única línea, indentándolo una vez (2 espacios). En este último caso, la coma de separación del último elemento también se escribirá, aunque no existan más elementos. De esta forma se evitan errores al añadir nuevos elementos al vector.

```
$vector1 = array(1, 2, 'clave' => 'valor');
$vector2 = array(
```

```
'forum' => 'foro1',  
'template' => 'forums',  
'arguments' => array('tid' => NULL, 'topics' => NULL),  
'size' => 128,);
```

3.3.9 Nombres de archivos

Los nombres de archivos deben escribirse siempre en minúsculas. La única excepción son los archivos de documentación, que tendrán extensión .txt y el nombre es mayúsculas. Por ejemplo, README.txt, INSTALL.txt, etc.

3.3.10 Comentar el código

En este apartado debemos diferenciar entre los comentarios para aclarar determinados fragmentos de código, que se insertan en cualquier punto de este, y los comentarios de documentación.

Los comentarios de documentación suelen escribirse al principio de un archivo o de cada función y se utilizan para generar documentación de ayuda a través de aplicaciones que extraen la información a partir de las etiquetas empleadas.

En el primero de los casos se pueden utilizar las etiquetas */* */* para comentarios en varias líneas y *//* para comentarios de una única línea. Se deben escribir frases completas, comenzándolas con mayúscula y terminándolas con un punto. En caso de que en el comentario se haga referencia a una constante, ésta deberá escribirse en mayúsculas (por ejemplo *TRUE* o *FALSE*).

3.4 Pantallas principales del portal web

Las pantallas principales del portal web del PCC son el resultado visual donde se define la apariencia de cada una de las páginas del portal web, ubicando en ellas todos los elementos que las componen. A continuación se muestra la pantalla principal del portal web, las pantallas restantes se podrán observar en el [Anexo 2](#):

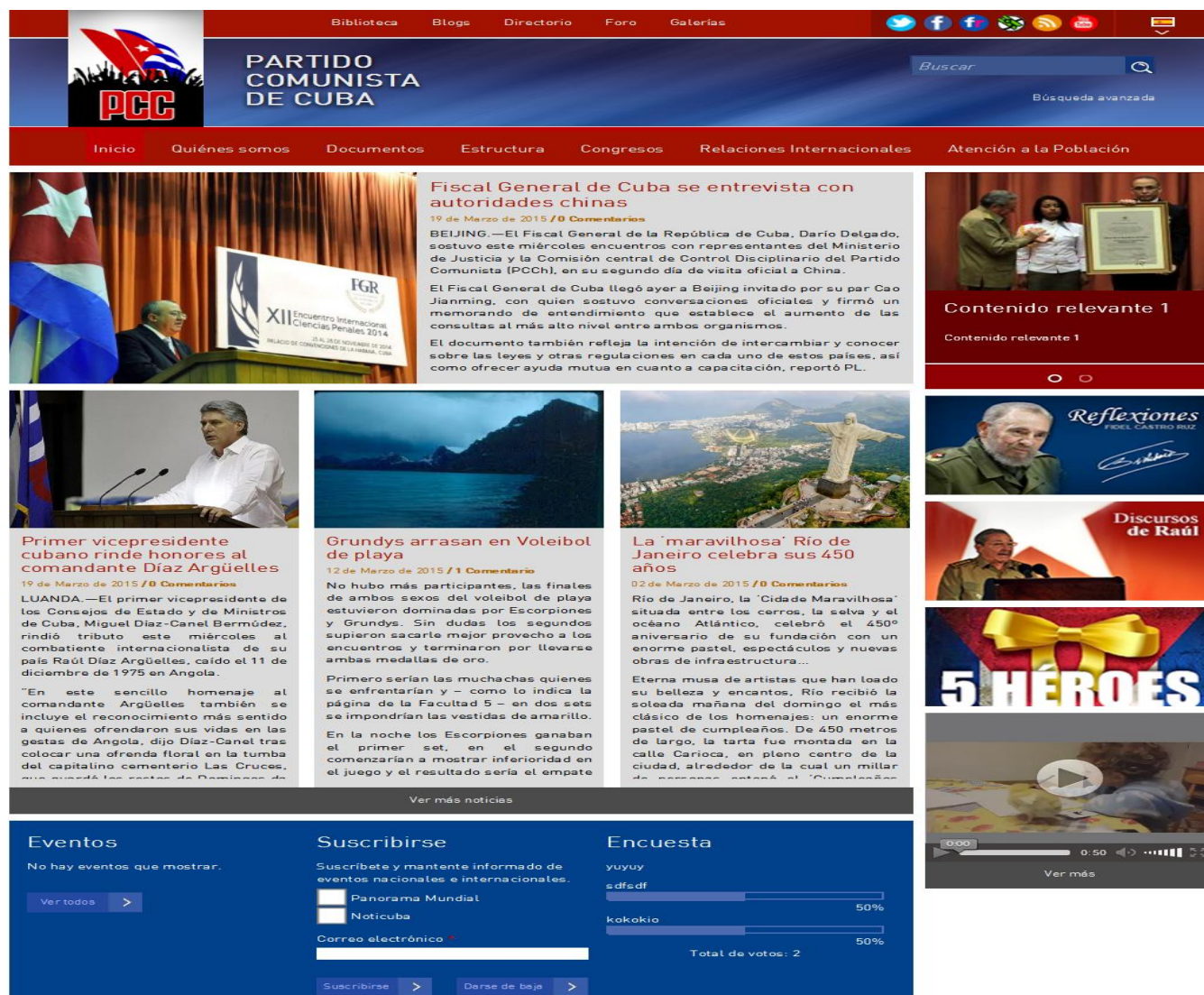


Figura 12. Pantalla principal del portal web del PCC

3.5 Validación del portal web

Las pruebas de *software* comprenden una fase muy importante del proceso de desarrollo. Tienen como objetivo fundamental determinar y asegurar la calidad, fiabilidad y robustez de un sistema, dentro del contexto o escenario donde está previsto que sea utilizado. Además permiten encontrar el mayor número de errores con la menor cantidad de tiempo y esfuerzo posible. Por otra parte muestran si las

funcionalidades del *software* operan según las especificaciones y requisitos definidos [37]. Existen varias clasificaciones de pruebas [38]:

- **Funcionalidad:** están enfocadas a realizar pruebas de funcionalidad, seguridad, y volumen. Estas pruebas son muy útiles ya que permiten comprobar los requisitos definidos contra las funcionalidades presentes en la aplicación, asegurar que el sistema es accedido por los actores definidos según niveles de acceso y verificar las habilidades de los programas para manejar grandes cantidades de datos.
- **Confiabilidad:** consisten en realizar pruebas de recuperación y tolerancia, además de las pruebas de *Benchmark* (comparativa). Permiten verificar que los procesos de recuperación restauran debidamente la base de datos, aplicaciones y sistemas. Además permite comparar el rendimiento de un elemento desconocido a uno de carga de trabajo de referencia conocido.
- **Usabilidad:** radican en realizar pruebas de usabilidad y de estructura. Comprueban factores humanos, estéticos, consistencia en la interfaz de usuario, ayuda sensitiva al contexto y en línea, asistente documentación de usuarios y materiales de entrenamiento. Además la valoración a la adherencia a su diseño y formación esto último es utilizado en las aplicaciones web para asegurar que todos los enlaces están conectados, el contenido deseado es mostrado y no hay contenido huérfano.
- **Eficiencia:** estas están dirigidas a realizar pruebas de contención, carga y estrés. En general estas pruebas permiten validar las habilidades del sistema para manejar la demanda de múltiples actores sobre un mismo recurso. También validar y valorar la aceptabilidad de los límites operacionales de un sistema bajo carga de trabajo variable, mientras el sistema bajo prueba permanece constante y además evaluar cómo el sistema responde bajo condiciones anormales (extrema sobrecarga).
- **Portabilidad:** consiste en realizar pruebas de configuración y de instalación. Estas están enfocadas a asegurar que un sistema funcione y pueda ser instalado en diferentes configuraciones de hardware y *software*.

En la validación del portal web del PCC es necesario tener en cuenta varias características dependiendo del tipo de aplicación, además del objetivo que persiguen las pruebas a realizar. Después de analizar las propiedades del portal web se determinó que es necesario medir su reacción integral frente a diversas acciones que podrán efectuar los usuarios, para ello es conveniente la selección de las siguientes pruebas: pruebas de funcionalidad, usabilidad, seguridad, carga y estrés. A continuación se describen las pruebas seleccionadas y los resultados arrojados con su ejecución.

3.5.1 Pruebas funcionales

Las pruebas funcionales están centradas en comprobar que las funcionalidades descritas en el documento de requisitos del sistema se cumplen con la implementación realizada. A este tipo de pruebas también se les denomina pruebas de comportamiento o de caja negra, debido a que los analistas enfocan su atención a las respuestas del sistema de acuerdo a los datos de entrada y sus resultados en los datos de salida, los cuales se definen generalmente en los casos de prueba que se crean antes del inicio de las pruebas [39].

El proceso para ejecutar este tipo de pruebas es el siguiente [40]:

1. Analizar los requisitos y sus especificaciones.
2. Seleccionar entradas válidas y no válidas de acuerdo con las especificaciones.
3. Determinar las salidas esperadas para cada entrada.
4. Diseñar los casos de pruebas con las entradas seleccionadas.
5. Ejecutar los casos de prueba.
6. Comparar las salidas encontradas con las salidas esperadas.
7. Determinar si el funcionamiento del *software* en prueba es apropiado.

A continuación se muestra el diseño de casos de pruebas de la funcionalidad RF1_Autenticar usuario. Se exponen otros ejemplos en el [Anexo 3](#), los casos de pruebas restantes se encuentran en el documento Diseño de casos de pruebas basado en requisitos del portal web del PCC.

Condiciones de ejecución: La página del portal web debe estar disponible.

SC RF1_Autenticar usuario

Escenario	Descripción	Nombre de Usuario	Contraseña	Respuesta del sistema	Flujo central
EC1.1 Autenticar usuario de forma correcta.	El sistema autentica al usuario de forma correcta.	admin	Admin123	El sistema autentica al usuario y le permite el acceso a las funcionalidades del sistema, según su rol.	El usuario accede a la página principal del portal web a la siguiente URL: URL/user. 1.-El sistema habilita un formulario para introducir la información.

					2.-El usuario introduce la información y presiona el botón: Iniciar sesión.
EC1.2 Autenticar usuario de forma incorrecta	El sistema no autentica al usuario	Admin	Admin123	El sistema no autentica al usuario y emite el mensaje de error: Lo sentimos. No reconocemos el nombre de usuario o la contraseña. ¿Olvidó su contraseña?	El usuario accede a la página principal del portal web a la siguiente URL: URL/user. 1.-El sistema habilita un formulario para introducir la información. 2.-El usuario introduce la información y presiona el botón: Iniciar sesión.
EC1.3 Autenticar usuario dejando campos vacíos	El sistema no autentica al usuario	vacío	Admin123	El sistema no autentica al usuario y emite los siguientes mensajes de errores: El campo Nombre de usuario es obligatorio. El campo Contraseña es obligatorio. Lo sentimos. No reconocemos el nombre de usuario o la contraseña. ¿Olvidó su contraseña?	El usuario accede a la página principal del portal web a la siguiente URL: URL/user. 1.-El sistema habilita un formulario para introducir la información. 2.-El usuario introduce la información y presiona el botón: Iniciar sesión.
		admin	vacío		

Tabla 3. Caso de prueba Autenticar usuario

Descripción de Variables

No	Nombre de campo	Clasificación	Valor Nulo	Descripción
1	Nombre de usuario	Campo de texto	No	Permite solo los caracteres (a-z, 0-9, ., _) No se permiten las cadenas compuestas Solamente por los caracteres (., _), estos no se permiten al inicio y al final de la cadena, ni uno al lado del otro.
2	Contraseña	Campo de texto	No	Permite todos los caracteres.

Tabla 4. Descripción de variables caso de prueba Autenticar usuario

3.5.2 Pruebas de usabilidad

Las pruebas de usabilidad permiten realizar un análisis destinado a identificar y rectificar las deficiencias de usabilidad y errores existentes, que los usuarios pueden encontrar al usar una aplicación. Tiene como objetivo asegurar que la aplicación sea fácil de aprender y usar, que cumpla con las funcionalidades para las cuales fue desarrollada. Aportan datos cuantitativos y cualitativos sobre usuarios reales que realizan tareas reales sobre una aplicación [41].

3.5.3 Pruebas de carga

Las pruebas de carga permiten la simulación del acceso de muchos usuarios a un servidor al mismo tiempo, posibilitando observar el comportamiento de una aplicación bajo una cantidad de peticiones esperadas. La carga pueden ser el número de usuarios concurrentes que se espera que utilicen la aplicación, y que realizan durante el tiempo en que dura la carga un número específico de transacciones. Este tipo de prueba facilita la monitorización del servidor y la base de datos para obtener el cuello de botella en la aplicación, y puede mostrar los tiempos de respuesta de todas las transacciones importantes [42].

3.5.4 Pruebas de estrés

Las pruebas de estrés posibilitan la obtención de datos sobre la carga del sistema. Tiene como objetivo generar cargas en el sistema hasta hacerlo inutilizable, para centrarse en verificar la calidad de los mensajes de error o establecer alertas para anticipar los fallos. Las pruebas de estrés son uno de los últimos tipos de pruebas que se deben efectuar, debido a que tienen un carácter poco realista ya que podría darse el caso que nunca se diera en la vida real la situación de carga simulada [43].

3.5.5 Pruebas de seguridad

Las pruebas de seguridad permiten realizar una evaluación de los sistemas desde el punto de vista externo y sin conocimiento previo de este. 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 [44].

3.5.6 Resultados de las pruebas

Resultados de las pruebas funcionales

Las pruebas funcionales aplicadas a la propuesta de solución son basadas en el diseño de casos de pruebas, donde se recogen los escenarios correspondientes a cada requisito funcional especificado. Para la realización de las pruebas se ejecuta cada caso de prueba utilizando datos válidos e inválidos y se eligen correctamente los valores de entrada abarcando el mayor número posible de combinaciones, sin que sea muy elevado el número de casos de pruebas.

En la primera iteración se detectaron 17 no conformidades, de ellas 10 no conformidades son de error de idioma, 2 no conformidades de validación en el campo usuario en los RF2 y RF3 y 5 no conformidades de errores funcionales en los RF6 y RF7. En la segunda iteración se encontró un total de 3 no conformidades de error de idioma. Todas las no conformidades encontradas se solucionaron por el equipo de desarrollo como se muestra en el siguiente gráfico.

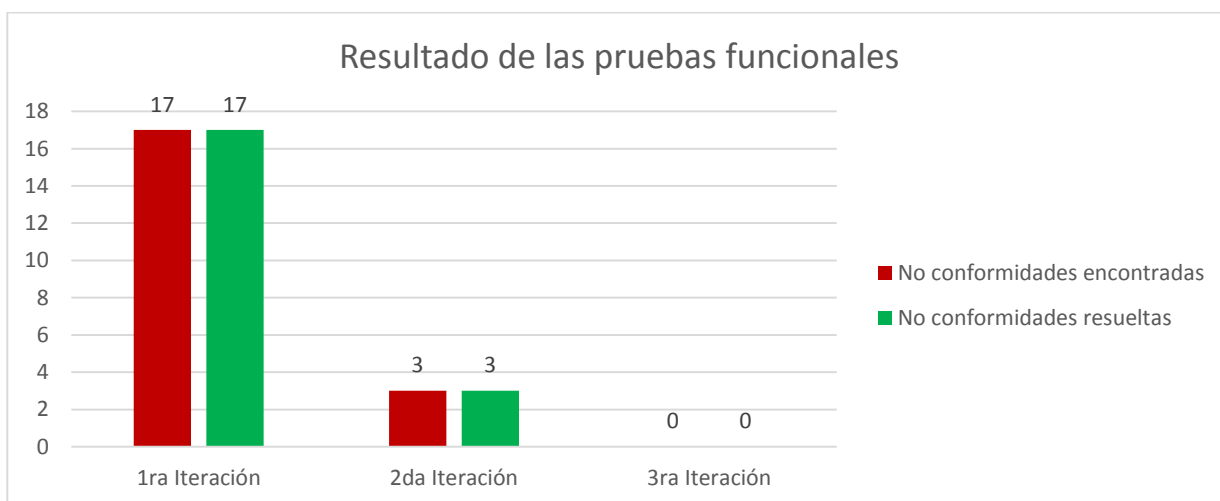


Gráfico 1. Resultados de las pruebas funcionales

Resultados de las pruebas de usabilidad

Las pruebas de usabilidad aplicadas a la propuesta de solución se realizaron a través del documento “Lista de chequeo de usabilidad de sitios web”. Esta cuenta con 110 indicadores separados por 13 elementos de usabilidad, la cual se observa en el [Anexo 4](#).

Se efectuaron dos iteraciones. En la primera iteración quedó evaluada de regular la prueba, debido a que se encontraron 2 no conformidades críticas y 1 no conformidades no críticas, para un total de 3 no conformidades, con 3 elementos afectados. En la segunda iteración se solucionaron todas las no conformidades, resultando satisfactoria la prueba.

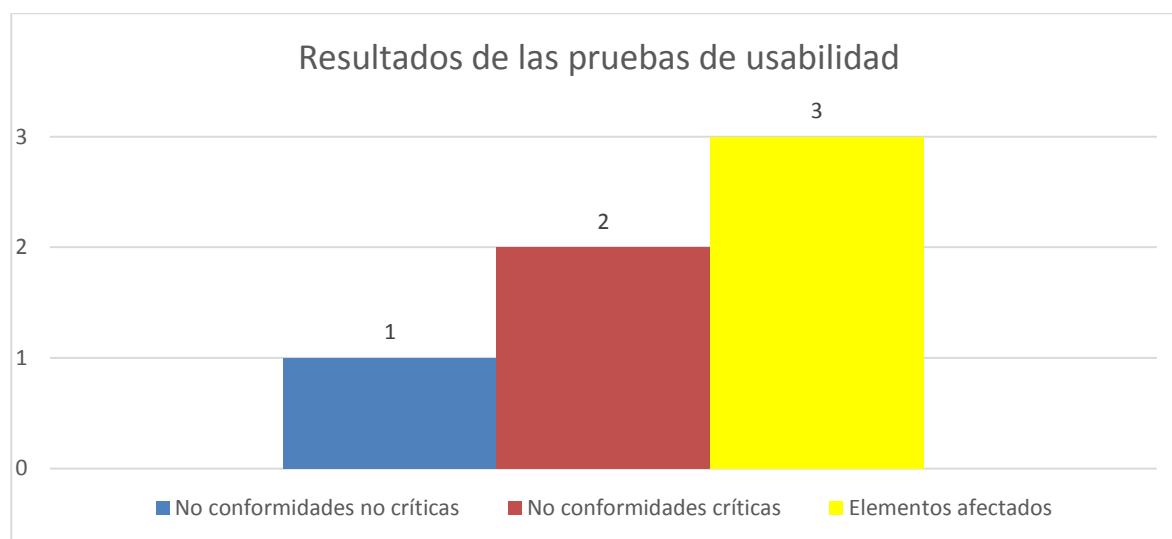


Gráfico 2. Resultado de las pruebas de usabilidad

Resultados de las pruebas de carga y estrés

Las pruebas de carga y estrés se desarrollaron con la ayuda de la herramienta *Apache JMeter 2.3.1*⁹, la cual permite medir el rendimiento de recursos como servicios web, lenguajes dinámicos y bases de datos, entre otros.

Estas se llevaron a cabo en un ambiente seleccionado utilizando un ordenador con las siguientes características:

- Sistema operativo Ubuntu.
- Microprocesador Intel Core i3 2.27GHz.
- Memoria RAM 4GB.

⁹ <http://jmeter.apache.org/>

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 que se especificó en los requerimientos del sistema. La propuesta de solución generó un total de 236952,3 kb/seg de transferencia de datos para 100 usuarios concurrentes esperados, lo que incurrió en un rendimiento de 2,8/seg. Se demuestra que la propuesta de solución es estable, ya que se mantuvo prestando servicios todo el tiempo sin incurrir en fallos. Además se probó para un total de 50 y 100 usuarios concurrentes, obteniendo los resultados que se muestran en el siguiente gráfico:

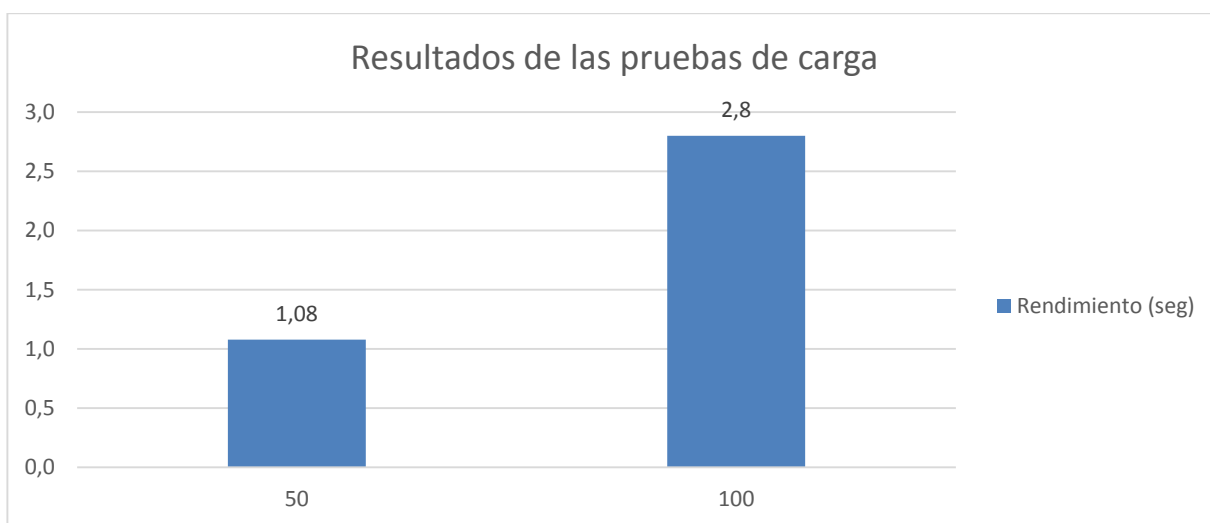


Gráfico 3. Resultados de las pruebas de carga

Los resultados de las pruebas de estrés se consideran satisfactorios, debido a que, después de sobrepasar la cantidad de 100 usuarios con un total de 150 y 200 usuarios concurrentes, la propuesta de solución se mantuvo estable prestando servicio todo el tiempo sin incurrir en fallos arrojando los resultados que se muestran en el siguiente gráfico:

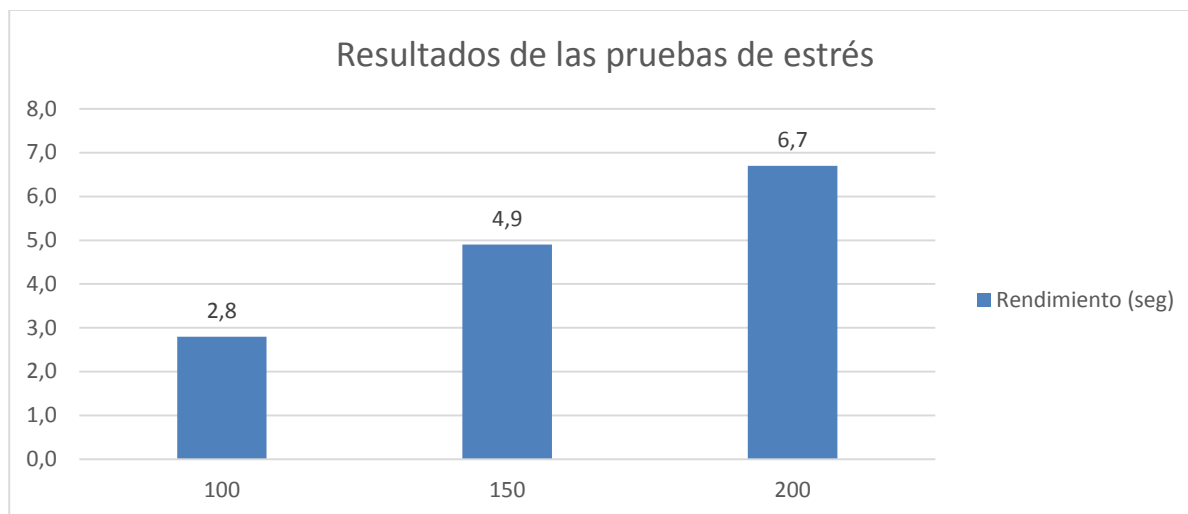


Gráfico 4. Resultados de las pruebas de estrés

Resultados de las pruebas de seguridad

Las pruebas de seguridad se aplicaron hasta el nivel 2. En el nivel 1 se utilizó la lista de chequeo ajustable a aplicaciones web, que consta de 15 indicadores divididos en 4 tipos de pruebas, la cual se observan en el [Anexo 5](#).

El nivel 1 en la primera iteración arrojó un total de 3 elementos afectados para 2 no conformidades críticas y 1 no conformidad no crítica quedando evaluada de insatisfactoria la prueba. Las no conformidades encontradas fueron solucionadas en su totalidad por el equipo de desarrollo. En la segunda iteración la prueba fue evaluada de satisfactoria, debido a que no se encontraron no conformidades. A continuación se muestran los resultados arrojados por las pruebas de seguridad de nivel 1.

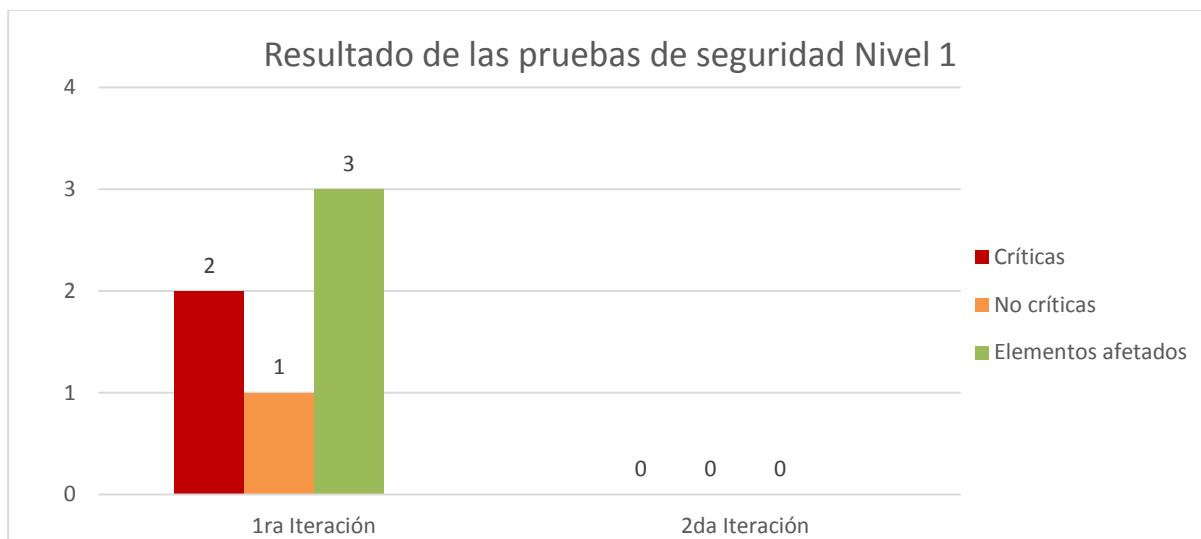


Gráfico 5. Resultados de las pruebas de seguridad de nivel 1

Se desarrollaron las pruebas de nivel 2 con la ayuda de la herramienta *Acunetix Web Vulnerability Scanner* ¹⁰, que establece alertas de tipo: alta, media, baja e informacional. En la primera iteración se detectaron 5 no conformidades medias, 1 no conformidades bajas para un total de 6 no conformidades y el equipo de desarrollo fue capaz de darle solución a estas. Se consideran de mayor importancia las no conformidades altas y medias. Las no conformidades referentes al entorno de despliegue de la aplicación son descartadas por el equipo de desarrollo porque existen entidades encargadas de solucionarlas. A continuación se muestran los resultados arrojados por las pruebas de seguridad de nivel 2.

¹⁰ <https://www.acunetix.com/vulnerability-scanner/>

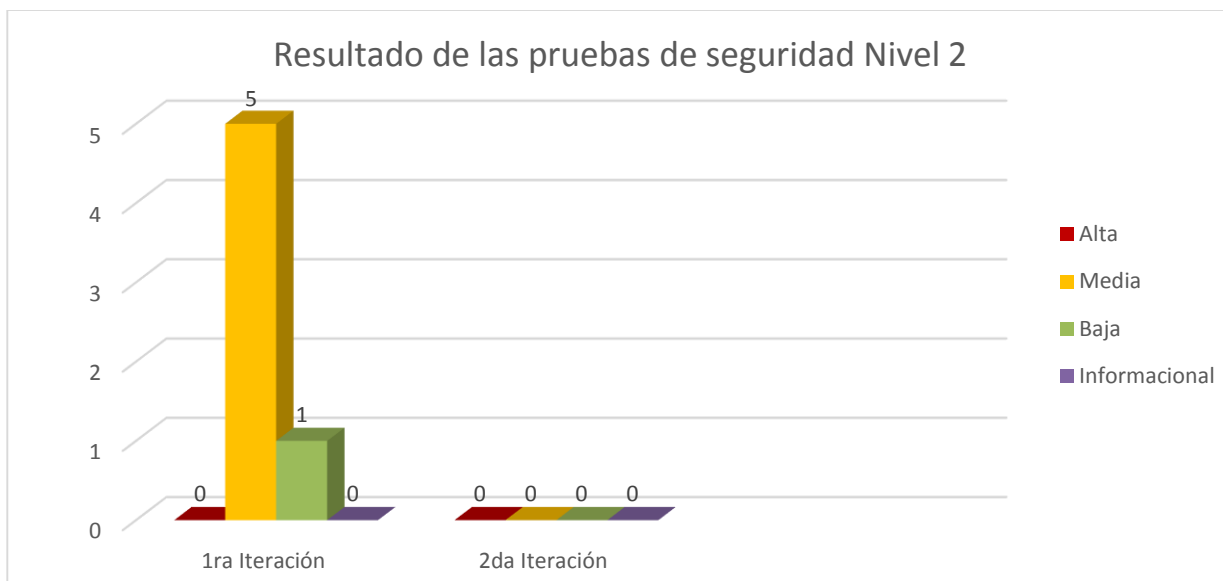


Gráfico 6. Resultados de las pruebas de seguridad de nivel 2

3.6 Conclusiones parciales

- ✓ En este capítulo la confección del diagrama de componentes ofreció una vista arquitectónica 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 pueda ser mantenido fácilmente, así como reutilizado por otros desarrolladores que lo necesiten.
- ✓ Por otra parte la realización de las pruebas de *software* al portal web facilitó encontrar los errores existentes para posteriormente solucionarlos, posibilitando determinar y asegurar la robustez del portal web del PCC.

Conclusiones

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

- ✓ La caracterización de las tendencias actuales de los sitios web dedicados a brindar información referida a partidos políticos, demostró la necesidad de desarrollar un portal web que fuese capaz de automatizar los procesos de gestión de los contenidos para el portal web del PCC.
- ✓ La definición de los requisitos funcionales y no funcionales facilitaron el análisis y diseño de las funcionalidades desarrolladas para el portal web del PCC.
- ✓ Los artefactos generados en el capítulo número 2 brindaron un mejor entendimiento de la propuesta de solución y constituyeron una guía fundamental para el desarrollo exitoso del sistema.
- ✓ La validación del portal web se efectuó a través de la realización de diferentes pruebas de *software*, permitiendo asegurar la calidad y la robustez del portal web del PCC.
- ✓ La metodología de desarrollo de *software* ágil *AUP* guió todo el ciclo de vida de la implementación del portal web, garantizando el óptimo desarrollo, logrando la documentación necesaria desde el comienzo y facilitando su estudio por parte de futuros miembros del equipo de desarrollo.

Recomendaciones

Para una futura versión del portal web del PCC se recomienda:

- ✓ Añadir una nueva funcionalidad que permita cargar las imágenes directamente desde *Flickr* para utilizarlas en los contenidos del sitio y en las propias galerías de imágenes o de infografías, como mismo se hace con los videos que se cargan en las galerías de videos desde *YouTube* y desde *TeVeo*.
- ✓ Añadir la opción de cargar los videos desde *Vimeo*, ya que existen módulos que lo permiten para aumentar las opciones de los videos que se utilizan en los contenidos del portal.
- ✓ Se recomienda compartir automáticamente los contenidos creados en las redes sociales previamente escogidas por el usuario.

Referencias Bibliográficas

1. Definición abc. *definicionabc.com*. [En línea] 2012. [Citado el: 22 de Enero de 2015.] <http://www.definicionabc.com/tecnologia/internet.php>.
2. **Mona Rusa, Francisco**. Programación en castellano. *programacion.com*. [En línea] 2007. [Citado el: 20 de Enero de 2015.] <http://www.programacion.com/html/articulo/gestioncontenidos/>.
3. Ve Multimedios. *vemultimedios.org*. [En línea] 2 de Abril de 2010. [Citado el: 12 de Noviembre de 2014.] <http://www.vemultimedios.org/espanol>.
4. **Centro de Apoyo Tecnológico a Emprendedores y otros**. *Estudio de los sistemas de gestión de Contenidos*. 2012. pág. 8.
5. Scribd. *scribd.com*. [En línea] 12 de Diciembre de 2009. [Citado el: 02 de Diciembre de 2014.] <http://es.scribd.com/doc/24012430/CMS-Sistema-de-Administracion-de-Contenidos#scribd>.
6. **Muras, Miguel Angel**. Mosaic. *mosaic.uoc.edu*. [En línea] 23 de Diciembre de 2009. [Citado el: 16 de Noviembre de 2014.] <http://mosaic.uoc.edu/2009/12/23/una-aproximacion-a-los-gestores-de-contenidos-cms/>.
7. Drupal Hispano. *drupal.org.es*. [En línea] 2012. [Citado el: 02 de Diciembre de 2015.] <http://drupal.org.es/caracteristicas>.
8. Joomlaos. *joomlaos.net*. [En línea] 2009. [Citado el: 17 de Noviembre de 2014.] <http://www.joomlaos.net/caracteristicas-de-joomla.php>.
9. **Angel Alvarez, Miguel**. *Introducción a los Lenguajes del Web*. 2012. pág. 16.
10. **Lorena Suárez, María**. *Programación en diversos lenguajes*. 2012. pág. 1.
11. Programación Web. *programacion-web-itcj.blogspot.com*. [En línea] 15 de Abril de 2015. [Citado el: 16 de Mayo de 2015.] <http://programacion-web-itcj.blogspot.com/2015/04/iv-programacion-del-lado-del-cliente-y.html>.

12. Definición abc. *definicionabc.com*. [En línea] 2010. [Citado el: 22 de Noviembre de 2014.] <http://www.definicionabc.com/tecnologia/html.php>.
13. Definición.de. *definicion.de*. [En línea] 2010. [Citado el: 02 de Diciembre de 2014.] <http://definicion.de/xml/>.
14. **Eguiluz, Javier**. *Introducción a JavaScript*. 2009. pág. 7.
15. **W3C**. *w3.org*. [En línea] 2012. [Citado el: 20 de Noviembre de 2014.] <http://www.w3.org/html/>.
16. KDE Documentation. *docs.kde.org*. [En línea] 2003. [Citado el: 21 de Noviembre de 2014.] <https://docs.kde.org/stable4/es/kdesdk/umbrello/uml-basics.html#about-uml>.
17. EcuRed. *ecured.cu*. [En línea] 2005. [Citado el: 22 de Noviembre de 2014.] http://www.ecured.cu/index.php/Sistema_Gestor_de_Base_de_Datos.
18. PostgreSQL-es. *postgresql.org*. [En línea] 02 de Octubre de 2010. [Citado el: 15 de Noviembre de 2014.] http://www.postgresql.org.es/sobre_postgresql.
19. NetBeans. *netbeans.org*. [En línea] 2012. [Citado el: 24 de Noviembre de 2014.] https://netbeans.org/index_es.html.
20. **Sánchez, Jorge**. *Servidores de Aplicaciones Web*. 2012. pág. 15.
21. **Márquez, José y otros**. *Un servidor Web gratis*. 2002. págs. 10-23.
22. **Knowledge Reuse Group**. *ie.inf.uc3m.es*. [En línea] 2013. [Citado el: 01 de Diciembre de 2014.] <http://www.ie.inf.uc3m.es/grupo/docencia/reglada/ls1y2/PracticaVP.pdf>.
23. **Carrillo Pérez, Isaias**. *Metodología de Desarrollo del Software*. 2008, pág. 3.
24. **Díaz Flores, Miriam Milagros**. *usmp.edu.pe*. [En línea] 07 de Octubre de 2008. [Citado el: 10 de Noviembre de 2014.] <http://www.usmp.edu.pe/publicaciones/boletin/fia/info49/articulos/RUP%20vs.%20XP.pdf>.
25. Extreme Programmin. *extremeprogramming.org*. [En línea] 2009. [Citado el: 10 de Noviembre de 2014.] <http://www.extremeprogramming.org>.

26. **Eclipse.** *eclipse.org*. [En línea] Agosto de 2008. [Citado el: 10 de Noviembre de 2014.] <http://www.eclipse.org/epf/general/OpenUP.pdf>.
27. Ingeniería de Software. *ingenieriadesoftware.mex.tl*. [En línea] 2015. [Citado el: 04 de Abril de 2015.] http://ingenieriadesoftware.mex.tl/63758_AUP.html.
28. **Perez Escobar, Carlos Javier.** *asprotech.blogspot*. [En línea] 10 de Octubre de 2013. [Citado el: 10 de Abril de 2015.] <http://asprotech.blogspot.com/2013/10/que-es-cmmi.html>.
29. Excriba. *excriba.prod*. [En línea] 2014. [Citado el: 20 de Enero de 2015.] <http://excriba.prod.uci.cu/page/document-details?nodeRef=workspace://SpacesStore/310f4da1-68a3-44a6-b9d4-2999a0c93a26>.
30. **Larman, Craig.** *UML y Patrones*. Segunda Edición. s.l. : Prentice Hall, 2002. págs. 97-98.
31. Tutorial Drupal. *cursosdrupal.com*. [En línea] 2012. [Citado el: 10 de Marzo de 2015.] <http://www.cursosdrupal.com/content/arquitectura>.
32. **Martin, Robert C.** *dsi.fceia.unr.edu.ar*. [En línea] 2009. [Citado el: 20 de Febrero de 2015.] www.dsi.fceia.unr.edu.ar/downloads/informatica/info_III/patrones.ppt.
33. **Gamma, Herich.** *Patrones de Diseño*. 2005, pág. 8.
34. **Rivera Alva, Eduardo.** *scribd.com*. [En línea] 11 de Noviembre de 2008. [Citado el: 03 de Marzo de 2015.] <http://es.scribd.com/doc/7884665/Arquitectura-de-Software-II-Diagrama-de-Componentes-y-Despliegue>.
35. **Gil, Fran.** *Experto en Drupal 7 Nivel Avanzado Curso de creación y gestión de portales web con Drupal 7*. 2012. págs. 59-60.
36. —. *Experto en Drupal 7 Nivel Avanzado Curso de creación y gestión de portales web con Drupal 7*. 2012, págs. 22-27.
37. **Mateo Navarro, Pedro Luis y otros.** *redalyc.org*. [En línea] Enero de 2009. [Citado el: 01 de Abril de 2015.] <http://www.redalyc.org/articulo.oa?id=92217159003>.

38. **Fables Estrada, Ailyn.** RCCI. [En línea] 2011. [Citado el: 2 de Mayo de 2015.] [http://rcci.uci.cu/index.php?journal=rcci&page=article&op=view&path\[\]=84&path\[\]=72](http://rcci.uci.cu/index.php?journal=rcci&page=article&op=view&path[]=84&path[]=72).
39. **Oré B., Alexander.** *calidadyssoftware.com*. [En línea] 2009. [Citado el: 13 de Abril de 2015.] http://www.calidadyssoftware.com/testing/pruebas_funcionales.php.
40. **Serna M., Edgar.** *bdigital.unal.edu.co*. [En línea] 2011. [Citado el: 14 de abril de 2015.] http://www.bdigital.unal.edu.co/6114/1/71617999._2012.pdf.
41. **Florián, Beatriz E. y otros.** *scielo.org.co*. [En línea] Julio de 2010. [Citado el: 10 de Mayo de 2015.] <http://www.scielo.org.co/pdf/eia/n13/n13a10.pdf>.
42. **Microsoft.** *msdn.microsoft.com*. [En línea] 2014. [Citado el: 13 de Abril de 2015.] <https://msdn.microsoft.com/es-es/library/ms182562%28v=vs.80%29.aspx>.
43. **Globe Testing.** *globetesting.com*. [En línea] 2011. [Citado el: 14 de Abril de 2015.] <http://www.globetesting.com/pruebas-de-rendimiento/>.
44. **V&V Quality.** *vyvquality.com*. [En línea] 2013. [Citado el: 20 de Abril de 2015.] http://www.vyvquality.com/w1/index.php?option=com_content&view=article&id=79&Itemid=151.

Bibliografía

Acunetix. *Web Application Security with Acunetix Web Vulnerability Scanner.* [En línea] 2015. <https://www.acunetix.com/vulnerability-scanner/>.

Angel Alvarez, Miguel. *Introducción a los Lenguajes del Web.* 2012. pág. 16.

Carrillo Pérez, Isaias. *Metodología de Desarrollo del Software.* 2008, pág. 3.

Centro de Apoyo Tecnológico a Emprendedores y otros. *Estudio de los sistemas de gestión de contenidos web.* 2012, pág. 8.

Definición abc. *definicionabc.com.* [En línea] 2012. [Citado el: 22 de Enero de 2015.] <http://www.definicionabc.com/tecnologia/internet.php>.

Definición abc. *definicionabc.com.* [En línea] 2010. [Citado el: 22 de Noviembre de 2014.] <http://www.definicionabc.com/tecnologia/html.php>.

Definición.de. *definicion.de.* [En línea] 2010. [Citado el: 02 de Diciembre de 2014.] <http://definicion.de/xml/>.

Díaz Flores, Miriam Milagros. *usmp.edu.pe.* [En línea] 07 de Octubre de 2008. [Citado el: 10 de Noviembre de 2014.] <http://www.usmp.edu.pe/publicaciones/boletin/fia/info49/articulos/RUP%20vs.%20XP.pdf>.

Eclipse. *eclipse.org.* [En línea] Agosto de 2008. [Citado el: 10 de Noviembre de 2014.] <http://www.eclipse.org/epf/general/OpenUP.pdf>.

EcuRed. *ecured.cu.* [En línea] 2005. [Citado el: 22 de Noviembre de 2014.] http://www.ecured.cu/index.php/Sistema_Gestor_de_Base_de_Datos.

Eguiluz, Javier. *Introducción a JavaScript.* 2009. pág. 7.

Excriba. *excriba.prod.* [En línea] 2014. [Citado el: 20 de Enero de 2015.] <http://excriba.prod.uci.cu/page/document-details?nodeRef=workspace://SpacesStore/310f4da1-68a3-44a6-b9d4-2999a0c93a26>.

Extreme Programmin. *extremeprogramming.org.* [En línea] 2009. [Citado el: 10 de Noviembre de 2014.] <http://www.extremeprogramming.org>.

Florián, Beatriz E. y otros. *scielo.org.co*. [En línea] Julio de 2010. [Citado el: 10 de Mayo de 2015.] <http://www.scielo.org.co/pdf/eia/n13/n13a10.pdf>.

Gamma, Herich. *Patrones de Diseño*. 2005, pág. 8.

Gil, Fran. *Experto en Drupal 7 Nivel Avanzado Curso de creación y gestión de portales web con Drupal 7*. 2012. págs. 59-60.

Gil, Fran. *Experto en Drupal 7 Nivel Avanzado Curso de creación y gestión de portales web con Drupal 7*. 2012, págs. 22-27.

Globe Testing. *globetesting.com*. [En línea] 2011. [Citado el: 14 de Abril de 2015.] <http://www.globetesting.com/pruebas-de-rendimiento/>.

González Cornejo, José Enrique. *¿Qué es UML?*. 2005.

Graf, Hagen y otros. *Joomla! 2.5 Guía para principiantes*. [En línea] 22 de Abril de 2012. [Citado el: 15 de Noviembre de 2014.]

http://comunidad.cav.uci.cu/index.php?option=com_phocadownload&view=category&id=27:manuales-y-tutoriales&Itemid=56.

Ingeniería de Software. *ingenieriadesoftware.mex.tl*. [En línea] 2015. [Citado el: 04 de Abril de 2015.] http://ingenieriadesoftware.mex.tl/63758_AUP.html.

Joomlaos. *joomlaos.net*. [En línea] 2009. [Citado el: 17 de Noviembre de 2014.] <http://www.joomlaos.net/caracteristicas-de-joomla.php>.

KDE Documentation. *docs.kde.org*. [En línea] 2003. [Citado el: 21 de Noviembre de 2014.] <https://docs.kde.org/stable4/es/kdesdk/umbrello/uml-basics.html#about-uml>.

Knowledge Reuse Group. *ie.inf.uc3m.es*. [En línea] 2013. [Citado el: 01 de Diciembre de 2014.] <http://www.ie.inf.uc3m.es/grupo/docencia/reglada/Is1y2/PracticaVP.pdf>.

Larman, Craig. *UML y Patrones*. Segunda Edición. s.l. : Prentice Hall, 2002. págs. 97-98.

Lorena Suárez, María. *Programación en diversos lenguajes*. 2012.

Marqués, Graells Pere. *LOS ESPACIOS WEB MULTIMEDIA: TIPOLOGÍA, FUNCIONES, CRITERIOS DE CALIDAD.* [En línea] 1999. [Citado el: 10 de Noviembre de 2014.] <http://peremarques.pangea.org/tipoweb.htm>.

Márquez, José y otros. *Un servidor Web gratis.* 2002. págs. 10-23.

Marticorena, Juan R. *Drupal 7 para usuarios y clientes.* [En línea] 25 de Noviembre de 2012. [Citado el: 16 de Noviembre de 2014.] http://comunidad.cav.uci.cu/index.php?option=com_phocadownload&view=category&id=29:manuales-y-tutoriales&Itemid=56.

Martin, Robert C. *dsi.fceia.unr.edu.ar.* [En línea] 2009. [Citado el: 20 de Febrero de 2015.] www.dsi.fceia.unr.edu.ar/downloads/informatica/info_III/patrones.ppt.

Mateo Navarro, Pedro Luis y otros. *redalyc.org.* [En línea] Enero de 2009. [Citado el: 01 de Abril de 2015.] <http://www.redalyc.org/articulo.oa?id=92217159003>.

Microsoft. *msdn.microsoft.com.* [En línea] 2014. [Citado el: 13 de Abril de 2015.] <https://msdn.microsoft.com/es-es/library/ms182562%28v=vs.80%29.aspx>.

Mona Rusa, Francisco. *Programación en castellano. programacion.com.* [En línea] 2007. [Citado el: 20 de Enero de 2015.] <http://www.programacion.com/html/articulo/gestioncontenidos/>.

Muras, Miguel Angel. *Mosaic. mosaic.uoc.edu.* [En línea] 23 de Diciembre de 2009. [Citado el: 16 de Noviembre de 2014.] <http://mosaic.uoc.edu/2009/12/23/una-aproximacion-a-los-gestores-de-contenidos-cms/>.

NetBeans. *netbeans.org.* [En línea] 2012. [Citado el: 24 de Noviembre de 2014.] https://netbeans.org/index_es.html.

Oré B., Alexander. *calidadyssoftware.com.* [En línea] 2009. [Citado el: 13 de Abril de 2015.] http://www.calidadyssoftware.com/testing/pruebas_funcionales.php.

Perez Escobar, Carlos Javier. *asprotech.blogspot.* [En línea] 10 de Octubre de 2013. [Citado el: 10 de Abril de 2015.] <http://asprotech.blogspot.com/2013/10/que-es-cmmi.html>.

Programación Web. *programacion-web-itcj.blogspot.com*. [En línea] 15 de Abril de 2015. [Citado el: 16 de Mayo de 2015.] <http://programacion-web-itcj.blogspot.com/2015/04/iv-programacion-del-lado-del-cliente-y.html>.

Rivera Alva, Eduardo. *scribd.com*. [En línea] 11 de Noviembre de 2008. [Citado el: 03 de Marzo de 2015.] <http://es.scribd.com/doc/7884665/Arquitectura-de-Software-II-Diagrama-de-Componentes-y-Despliegue>.

Sánchez, Jorge. *Servidores de Aplicaciones Web*. 2012. pág. 15.

Scrip. *scribd.com*. [En línea] 12 de Diciembre de 2009. [Citado el: 02 de Diciembre de 2014.] <http://es.scribd.com/doc/24012430/CMS-Sistema-de-Administracion-de-Contenidos#scribd>.

Serna M., Edgar. *bdigital.unal.edu.co*. [En línea] 2011. [Citado el: 14 de abril de 2015.] http://www.bdigital.unal.edu.co/6114/1/71617999._2012.pdf.

Sparx Systems. [En línea] 2011. [Citado el: 23 de 02 de 2015.] http://www.sparxsystems.com.ar/resources/tutorial/uml2_sequencediagram.html.

Tecnología y Synergix. [En línea] 10 de Julio de 2008. [Citado el: 19 de Febrero de 2015.] <https://synergix.wordpress.com/2008/07/10/modelo-de-dominio/>.

Tecnológico Albacete. *Estudio de los sistemas de gestión de contenidos web*. 2012.

Tutorial Drupal. *cursosdrupal.com*. [En línea] 2012. [Citado el: 10 de Marzo de 2015.] <http://www.cursosdrupal.com/content/arquitectura>.

Ve Multimedia. *vemultimedios.org*. [En línea] 2 de Abril de 2010. [Citado el: 12 de Noviembre de 2014.] <http://www.vemultimedios.org/espanol>.

Visconti, Marcello. *Fundamentos de la Ingeniería de Software*. 2010, pág. 19.

V&V Quality. *vyvquality.com*. [En línea] 2013. [Citado el: 20 de Abril de 2015.] http://www.vyvquality.com/w1/index.php?option=com_content&view=article&id=79&Itemid=151.

W3C. *w3.org*. [En línea] 2012. [Citado el: 20 de Noviembre de 2014.] <http://www.w3.org/html/>.