

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

FACULTAD 1



Título

Desarrollo de un nuevo portal web para el Centro de Información Científico Técnica de la Universidad de las Ciencias Informáticas

Autor: Oberman Eriza Oviedo

Tutores:

MSc. Laritza Magdalena Martínez Negrín

MSc. Lydia Iby Roa Contreras

Ing. Nodelvis Hernández Rodríguez

La Habana, junio 2016

Pensamiento



“Solamente puedes aprender si tú mismo te abres a diferentes fuentes de información.”

Frank Tankerton

Declaración de autoría

Declaro ser el único autor de la presente tesis y reconozco a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

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

Firma del Autor
Oberman Eriza Oviedo

Firma de Tutor
MSc. Laritza M. Martínez
Negrín

Firma de Tutor
MSc. Lydia Iby Roa Contreras

Firma de Tutor
Ing. Nodelvis Hernández
Rodríguez

Datos de contactos**Autor:**

Oberman Eriza Oviedo

Universidad de las Ciencias Informáticas (UCI)

La Habana, Cuba

e-mail: oyeriza@estudiantes.uci.cu

Tutores:

Mcs. Laritza Magdalena Martínez Negrín

Universidad de las Ciencias Informáticas (UCI)

La Habana, Cuba

e-mail: laritza@uci.cu

MSc. Lydia Iby Roa Contreras

Universidad de las Ciencias Informáticas (UCI)

La Habana, Cuba

e-mail: lydiaiby@uci.cu

Ing. Nodelvis Hernández Rodríguez

Universidad de las Ciencias Informáticas (UCI)

La Habana, Cuba

e-mail: nodelvis@uci.cu

Dedicatoria

A mis padres por todo el apoyo, el amor y la confianza depositada.

A mi hermana por su ayuda, entrega y por soportarme estos años.

A mis tíos, tías y primos por la preocupación y la ayuda durante la carrera.

Agradecimientos

A mis tutores Nodelvis, Laritza, Lydia y a Lisbet sin ellos hubiera sido imposible la realización de este trabajo.

A todos mis nuevos hermanos, los que terminaron y los que se fueron quedando en el camino, los cuales siempre van a ocupar un lugar importante en mi corazón: Darío, Rene, Manano, Rubén, David, Sánchez, Amaury, Duanny, Pablo, Michel, Yojahny, Raisa, Aimet, Lisbeth, Betsy, Iguelit, Carlos, Hosford, Yasiel y que me disculpen si deje a algunos afuera, los quiero también.

A mi equipo de futbol el Inter por su apoyo y ayuda en estos años, Nolberto, Eduardito, Pedro, Cesar, Angel, Zapato, el Micro y Alexo que por cierto mi herma ya quedaste para calentar banco, y no puede faltar la fans numero 1 la flaca Ilismay.

A todas aquellas personas que de una forma u otra hice de su apto como mío: Yosel, Yordy, Yosmay, Adrian, Justiz, Yadriel, Susana, Daniel, Maikel, Wilbia, Baby, Laura y los que no mencione que también me acogieron en sus apto.

Vuelvo y repito si deje a alguien afuera por favor discúlpenme, los quiero igual.

Resumen

La presente investigación se refiere a un problema de actualidad y gran importancia para los centros de información como es la gestión eficiente de sus recursos informativos. Para dar respuesta a tal situación se diseña un nuevo portal web para el Centro de Información Científico-Técnica (CICT) de la Universidad de las Ciencias Informáticas (UCI). El diseño y las funcionalidades del nuevo portal favorecerán la divulgación y acceso a los recursos de información que gestiona el CICT, cuya misión es apoyar procesos sustantivos como la docencia, la investigación, la extensión y la producción generados en la Universidad. Dicha propuesta tributa directamente en la satisfacción de las necesidades de información científico-técnica de los usuarios en la toma de decisiones y la planificación estratégica. Para el desarrollo de la propuesta, se emplearon metodologías y herramientas de código abierto, Drupal, PostgreSQL, NetBeans, Apache, PHP, CSS, HTML, *JavaScript* y la metodología de desarrollo Proceso Unificado Ágil (AUP, siglas en inglés *Agile Unified Process*) variación UCI de acuerdo con las políticas de informatización de la Universidad.

Palabras clave: portal web, recurso de información, recursos informativos, gestión de información

Índice

Introducción	1
Capítulo 1: La gestión de información en los portales web de los Centros de Información Científico Técnica	7
1.1. Conceptos fundamentales asociados al tema de investigación.....	7
1.2. Estudio de sistemas homólogos	8
1.3. Sistema Gestor de Contenidos	11
1.5. Servidores web	13
1.6. Lenguajes de desarrollo.....	14
1.7. Herramientas utilizadas en las pruebas	16
1.8. Entorno de desarrollo integrado.....	17
1.9. Herramientas CASE	17
1.10. Metodologías de desarrollo de <i>software</i>	18
Conclusiones parciales	19
Capítulo 2: Caracterización del portal web del Centro de Información Científico-Técnica de la UCI	20
2.1 Propuesta del sistema	20
2.2 Modelado conceptual.....	20
2.3 Levantamiento de requisitos	22
2.4 Historias de Usuario	31
2.5 Descripción de estilos arquitectónicos y patrones de diseño	35

2.6	Diagrama de Clases del Diseño.....	37
2.7	Modelo de Despliegue	38
	Conclusiones parciales	39
Capítulo 3: Implementación y validación del portal web para el Centro de Información Científico-Técnica de la UCI.....		41
3.1.	Diagrama de componente.....	41
3.2.	Estándares de codificación	42
3.3.	Validación de la propuesta de solución	44
	Conclusiones parciales	54
	Conclusiones generales.....	54
	Recomendaciones	55
	Referencias bibliográficas	56
	Bibliografía.....	59
	Glosario de términos.....	62

Índice de Ilustraciones

Ilustración 1: Modelo conceptual de la propuesta de solución.....	22
Ilustración 2: Arquitectura de la propuesta de solución.	36
Ilustración 3: Diagrama de clase del diseño.....	38
Ilustración 4: Diagrama del modelo de despliegue.....	39
Ilustración 5: Diagrama de componente.....	41
Ilustración 6: Ejemplo de Identación	43
Ilustración 7: Ejemplo de uso de punto y coma en código PHP.....	43
Ilustración 8: Ejemplo de uso de comillas	43
Ilustración 9: Ejemplo de estructuras de control.....	44
Ilustración 10: Comportamiento de las no conformidades por iteraciones.....	49
Ilustración 11: Resultado de las pruebas de carga y estrés.....	52

Índice de Tablas

Tabla 1: Indicadores de evaluación de los sistemas homólogos	9
Tabla 2: Requisitos funcionales	22
Tabla 3: Historia de Usuario. Autenticar usuario	31
Tabla 4: Historia de Usuario. Crear usuario	33
Tabla 5: Caso de prueba: Crear aviso	45
Tabla 6: Descripción de las variables de Crear aviso	46
Tabla 7: Caso de prueba: Crear vínculos bibliotecarios	46
Tabla 8: Descripción de las variables Crear vínculos bibliotecarios	48
Tabla 9: Resultado de la aplicación de la herramienta Acunetix WVS 8 para pruebas de seguridad	50
Tabla 10: Resultados de las pruebas de usabilidad	53

Introducción

La sociedad de la información es la expresión de las realidades y capacidades de los medios de comunicación más nuevos o novedosos merced al desarrollo tecnológico que se consolidó en la última década del siglo XX; llamada así además, debido al impacto de las tecnologías de la información en los procesos que realizan las unidades de información (archivos, bibliotecas, centros de información o documentación, entre otros).

En las unidades de información, la gestión de la información se vincula con la generación y la aplicación de estrategias, dada la importancia que posee la información como recurso indispensable para la toma de decisiones, la planificación estratégica, el establecimiento de políticas y el desarrollo de una cultura organizacional en función de los objetivos y metas de las organizaciones.

Estas organizaciones de información entre las que se pueden mencionar las bibliotecas, son locales que ofrecen condiciones adecuadas para la lectura, el estudio y la recreación, que contienen un acervo de libros catalogados y clasificados, organizados de tal forma que quienes deseen hacer uso de ellos puedan encontrarlos con facilidad y rapidez.

Otro tipo de unidad de información son los centros de información o documentación, cuya función radica en examinar y evaluar minuciosamente las fuentes primarias y secundarias de información técnica y científica; es decir, transformar fuentes primarias de información científica y técnica en productos que sean utilizados directamente por los usuarios del centro de documentación.

Para estos centros la gestión de información, según (Ponjuán, 2007), es “el proceso mediante el cual se obtienen, despliegan o utilizan recursos básicos (económicos, físicos, humanos, materiales) para manejar información dentro¹ y para la sociedad a la que sirve. Tiene como elemento básico la gestión del ciclo de vida de este recurso y ocurre en cualquier organización. Es propia también de unidades especializadas que manejan este recurso en forma intensiva, llamadas unidades de información”.

¹ Se refiere a la información generada por la unidad de información

Otro concepto asociado al de unidades de información es el de fuentes de información que son aquellas que contienen datos útiles para satisfacer una demanda de información o conocimiento y pueden ser primarias o secundarias (Biblioteca Universidad de Alcalá, 2015).

Las fuentes de información son de vital importancia para las unidades de información ya que estas nutren a los recursos informativos los cuales son toda la documentación de la organización, textual y no textual, externa o interna, formal o informal (Moscoso, 1998).

El auge de las Tecnologías de la Información y la Comunicación (en lo adelante TIC) y las ventajas que estas reportan al posibilitar el incremento de la socialización del conocimiento se evidencian en las universidades que cuentan con centros de información que brindan a los estudiantes y profesores toda una gama de recursos informativos que apoyan el aprendizaje, a los cuales se acceden a través de portales web, los cuales son un sitio web que ofrece de forma íntegra, una variedad de servicios al usuario, basándose en una misma temática y de forma organizada (Bravo y García 2014).

La utilización de portales web en las unidades de información otorga ventajas competitivas como son: mayor visibilidad y accesibilidad permanente a la información, promoción de los servicios, interacción con el usuario y funcionan como identificadores de información elaborada, resumida, estructurada, actualizada, organizada y precisamente por esto se valora el aporte significativo que hacen a la toma de decisiones y a la planificación estratégica de los centros de estudio, de investigación, etc.

El Centro de Información Científico-Técnica (CICT) de la Universidad de las Ciencias Informáticas (UCI), cuenta con un portal web como herramienta para la divulgación y acceso a los recursos informativos que apoyan la docencia, la investigación, la extensión universitaria y la producción.

Entre los recursos informativos que posee el portal web se encuentra un catálogo en línea en el cual se pueden realizar búsquedas simples y avanzadas de todas las colecciones de documentos existentes en el centro. De igual manera el portal facilita el acceso a bases de datos internacionales, así como a otros recursos de información en línea; enlaza recursos informativos como el Repositorio Institucional, el observatorio Vigitec y los boletines de los centros de desarrollo. La herramienta actual permite la divulgación de los boletines y revistas editadas por los especialistas del CICT y otras áreas de la UCI.

Sin embargo, a pesar de la situación descrita anteriormente, el portal presenta las siguientes deficiencias:

- No existe un buscador que permita recuperar la información publicada en el sitio, ajeno al catálogo del centro, lo cual provoca en el usuario pérdida de tiempo y agotamiento visual.
- No permite la interacción de los usuarios con los bibliotecarios, desaprovechando las potencialidades de la web 2.0.
- Predominan muchos espacios sin información y no permite al *webmaster* realizar cambios a la posición de algunos elementos para dinamizar el aspecto del portal.
- Se detectan determinados problemas que inciden en la organización y representación del contenido como la inexistencia de la categorización de los recursos de información, lo que provoca confusión en los usuarios. Ejemplo, el enlace al portal Vigitec se visualiza dentro del apartado de las publicaciones seriadas.
- No permite la publicación de más de dos noticias, impidiendo el flujo de información actualizada hacia los usuarios.
- No presenta un mapa del sitio para que los usuarios realicen la exploración de los recursos de información de forma expedita.

De lo analizado hasta aquí se puede resumir que la disponibilidad y organización de los recursos informativos propician una limitada gestión de información en el actual portal web del CICT de la UCI, lo cual conlleva a la disminución de su consulta por los usuarios.

La situación descrita conduce a formular el siguiente **problema de investigación**:

¿Cómo mejorar la gestión de información del portal web del Centro de Información Científico-Técnica de la Universidad de las Ciencias Informáticas?

El **objeto de estudio** de la presente investigación está dirigido al proceso de gestión de información de los portales web.

La presente investigación tiene como **objetivo general**: desarrollar un nuevo portal web para el CICT de la UCI que permita mejorar la gestión de información.

El **campo de acción** corresponde a la gestión de información del portal web del CICT de la UCI.

Como **objetivos específicos** se plantean:

1. Elaborar el marco teórico de la investigación mediante el estudio del estado del arte de sistemas web homólogos, herramientas y tecnologías más factibles para el desarrollo de la aplicación.
2. Realizar la implementación del portal web, a partir del levantamiento de requisitos, análisis y diseño del mismo.
3. Evaluar el correcto funcionamiento de la solución propuesta.

Tareas de investigación:

1. Establecimiento de los referentes teóricos fundamentales relacionados con el proceso de gestión de información en portales web para centros de información científico-técnica.
2. Identificación de los parámetros e indicadores de los portales web de centros de información de universidades nacionales y extranjeras.
3. Análisis de las tecnologías, herramientas, estándares, patrones y metodologías para el desarrollo del portal web para el Centro de Información Científico Técnica de la Universidad de las Ciencias Informáticas.
4. Análisis y diseño de las funcionalidades del portal web para el Centro de Información Científico Técnico de la Universidad de las Ciencias Informáticas.
5. Implementación de la propuesta de solución para perfeccionar la gestión de información del Centro de Información Científico Técnico de la Universidad de las Ciencias Informáticas.
6. Evaluación de la propuesta de solución para garantizar su correcto funcionamiento.

Para guiar la investigación se plantea la siguiente **idea a defender**: el desarrollo de un nuevo portal web para el Centro de Información Científico Técnico de la UCI permitirá una mejor gestión de información.

Los métodos de investigación que se utilizan son:

Métodos teóricos:

Analítico-Sintético: se emplean en todo el desarrollo de la investigación, en la delimitación de su objeto de estudio y campo de acción, en la selección de las mejores tecnologías, herramientas, estándares, patrones y metodologías a emplear, en el análisis de los problemas presentes en el portal web para el CICT de la UCI.

Métodos empíricos:

Análisis documental: se emplea en la revisión de la bibliografía asociada al tema de investigación. Se revisa el proceso de gestión de información, recursos informativos, fuentes de información, así como materiales relacionados con el estudio de parámetros e indicadores definidos para evaluar portales web, se incluyen las tecnologías, herramientas, estándares, patrones y metodologías para el desarrollo de la propuesta.

Entrevista: se realiza una entrevista no estructurada a modo de conversación con el cliente donde se identifican los recursos de información que brinda el portal web actual y los requisitos funcionales que van a pasar a formar parte de la propuesta.

Observación: este método fue utilizado para realizar un análisis visual y constatar el funcionamiento del portal web del Centro de Información Científico Técnica de la Universidad de las Ciencias Informáticas.

Esta investigación se encuentra estructurada de la siguiente manera:

Capítulo 1: “La gestión de información en los portales web de los Centros de Información Científico Técnica”, En el presente capítulo se describen los principales conceptos para comprender el dominio del problema y un estudio de sistemas homólogos analizando diferentes parámetros e indicadores. Además, se describen las tendencias, metodologías, tecnologías y herramientas que se emplean en la implementación de la propuesta de solución.

Capítulo 2: “Caracterización del portal web del Centro de Información Científico-Técnica de la UCI.” En este capítulo se explica cómo mediante la realización del modelado de diagramas se desarrolla el flujo actual de los procesos, además se describe la propuesta de solución para resolver el problema planteado,

especificando los requisitos funcionales y no funcionales, los elementos fundamentales del diseño y la arquitectura que se debe tener en cuenta para el desarrollo del sistema.

Capítulo 3: “Implementación y validación del portal web para el Centro de Información Científico-Técnica de la UCI”. Se exponen algunos aspectos asociados con la implementación de la solución informática, así como los componentes que la integran. Además, se presentan los diseños de casos de prueba a utilizar en la validación del sistema y se analizan los resultados de las pruebas realizadas que permiten evaluar la calidad de la propuesta de solución.

Capítulo 1: La gestión de información en los portales web de los Centros de Información Científico Técnica

Este capítulo contiene los fundamentos teóricos que sustentan la gestión de información realizada en los portales web para centros de información científico-técnica. En el mismo se hace un estudio de sistemas homólogos, analizando diferentes parámetros e indicadores. Además, se describen los principales conceptos para comprender el dominio del problema, así como las tendencias, metodologías, tecnologías y herramientas que se emplean en la implementación de la propuesta de solución.

1.1. Conceptos fundamentales asociados al tema de investigación

Algunos conceptos necesarios para comprender la esencia de la presente investigación se exponen a continuación.

Según Gloria Ponjuán la **gestión de información** se definiría como “el proceso mediante el cual se obtienen, despliegan o utilizan recursos básicos (económicos, físicos, humanos, materiales) para manejar información dentro y para la sociedad a la que sirve. Tiene como elemento básico la gestión del ciclo de vida de este recurso y ocurre en cualquier organización. Es propia también de unidades especializadas que manejan este recurso en forma intensiva, llamadas unidades de información”. El proceso de gestión de información debe ser valorado sistémicamente en diferentes dimensiones y el dominio de sus esencias permite su aplicación en cualquier organización (Ponjuán, 2007).

Por su parte (Choo, 2002) plantea que la **gestión de información** tiene como objetivo optimizar la utilidad y contribución de los recursos de información con el fin de alcanzar los objetivos de la organización. En este sentido, la práctica de la gestión de información se traduce en la creación de canales y medios para transmitir y acceder a la información, así como añadirle valores a ésta.

En el glosario *Harrod's Librarians' Glossary and Reference Book*, la **gestión de la información** se define como un "término impreciso" que sirve para designar un conjunto de actividades orientadas a la generación, coordinación, almacenamiento o conservación, búsqueda y recuperación de la información tanto interna como externa contenida en cualquier soporte (Prytherch, 2000).

Para la presente investigación a partir de las fuentes consultadas, el autor determina seguir la definición establecida por Ponjuán por ser concisa y abarcadora ya que explica resumidamente el proceso del ciclo de vida de la información desde que llega y es procesada para diferentes organizaciones, que no son abordados por los otros autores.

En relación con la conceptualización de **portal web**, según Juan C. García Gómez, es un punto de entrada a Internet donde se organizan sus contenidos, ayudando al usuario y concentrando servicios y productos (Gómez, 2011).

Bravo y García definen un **portal web** como un sitio web que ofrece de forma íntegra, una variedad de servicios al usuario, basándose en una misma temática y de forma organizada (Bravo y García 2014).

Sobre la base de las anteriores definiciones el autor de la presente investigación considera que un **portal web** es una página diseñada para servir como punto de partida en la navegación, que brinda distintos servicios mediante la gestión de información.

Purificación Moscoso identifica el **recurso de información** como toda la documentación de la organización, textual y no textual, externa o interna, formal o informal, ... permiten a las organizaciones comunicarse con su entorno e influir en él, tratando de alcanzar unos objetivos previstos (Moscoso, 1998).

Según (Burk y Horton, 1991) los **recursos de información** son una configuración de personas, *software*, *hardware*, material, espacio, información y otros recursos de entrada que se describen ampliamente como funciones de almacenamiento y manejo de información, o, alternativamente, como fuentes de información, servicios y sistemas.

El autor, apoyándose en las definiciones anteriores cree que los recursos de información son el conjunto de fuentes, servicios, productos y sistemas de información previamente almacenados y/o administrados que permiten adquirir, ampliar, precisar o comunicar información y conocimientos que pueden ser utilizadas para satisfacer necesidades informativas.

1.2. Estudio de sistemas homólogos

Los portales web de los centros de información universitarios constituyen uno de los recursos que con mayor impacto marcan la diferencia que están experimentando estas unidades de información en su

transición hacia la era digital. Para el estudio de diferentes portales en Cuba y el resto del mundo se seleccionan un conjunto de parámetros con sus correspondientes indicadores.

A continuación se detallan algunos parámetros e indicadores de la metodología de evaluación de recursos digitales del profesor Lluís Codina,(Codina 2007), donde los parámetros responden a la pregunta: ¿qué queremos evaluar? y los indicadores responden a la pregunta: ¿cómo evaluamos este parámetro?

Tabla 1: Indicadores de evaluación de los sistemas homólogos

Parámetros	Indicadores
Autoría	Autoría del recurso bien determinada.
	Posibilidad de contactar con el autor o institución.
Navegación y recuperación	Mapa de navegación o sumario.
	Sistema de búsqueda.
Ergonomía	Tipografía adecuada.
	Visualización agradable.

1.2.1. Portal web del CICT de la Universidad de Jaén

El portal web del CICT de la Universidad de Jaén (<https://gsya.ujaen.es/>) presenta la posibilidad de contactar con la institución, pero no se puede demostrar la autoría del recurso, sistemas de búsquedas y mapa de navegación debido a que el portal restringe el acceso solo a los usuarios de la Universidad de Jaén, aunque presenta una tipografía adecuada, los contenidos no son visualizados correctamente por la combinación de colores de las letras con el fondo.

1.2.2. Portal web del Centro de Recursos de Aprendizaje y la Investigación (CRAI) de la Universidad de Sevilla

El portal web del CRAI de la Universidad de Sevilla (<http://bib.us.es/ulloa/>) contiene posibilidad de contacto con la institución y autoría en los recursos. Muestra un sistema de búsqueda de la información con una pobre navegación por la falta del mapa de navegación. No obstante, la tipografía es adecuada con una visualización agradable.

1.2.3. Portal web del CRAI de la Universidad de Cienfuegos

EL portal web del CRAI de la Universidad de Cienfuegos (<http://crai.ucf.edu.cu/>) presenta información de contacto con el autor del sitio, pero no de los recursos. Aunque el sitio tiene una tipografía adecuada y una agradable visualización, la navegación y la búsqueda en el sitio se torna complicada.

1.2.4. Portal web del CICT de la Universidad de Matanzas

El portal web del CICT de la Universidad de Matanzas (<http://cict.umcc.cu/>) permite el contacto con la institución, además están bien definida la autoría de los recursos que brinda el sitio. La navegación en el sitio es engorrosa debido a que no presenta un mapa del sitio y los enlaces están distribuidos incorrectamente, presenta un sistema de búsqueda para los recursos, pero no tiene uno general. La tipografía es adecuada pero la visualización no es agradable debido a una mala arquitectura de información.

Resumen de los sistemas homólogos estudiados

Los portales web estudiados brindaron información referente a la autoría, la ergonomía, la navegación y recuperación. Cada uno cuenta con funcionalidades que cubren sus propias necesidades, en algunos casos son muy similares a las que se necesitan establecer ya que están encaminadas a la capacidad de interactuar, solicitar documentos, descargar archivos, consultar y leer artículos de revistas, navegar de un enlace a otro hasta encontrar lo que necesitan, tanto en el ámbito de alcance del centro consultado como en otros diseminados por todo el mundo.

Debido a que no van dirigidos a los mismos usuarios y las interfaces no responderían a los intereses del cliente, la reutilización de contenidos se dificulta, provocando que los procesos de publicación y

mantenimiento del sitio se vuelvan engorrosos, por tanto solo se pueden tomar de referencia haciéndose necesario el diseño e implementación de una solución propia para la UCI.

1.3. Sistema Gestor de Contenidos

Los sistemas gestores de contenidos (CMS en inglés *Content Management System*) son una especie de herramienta que permite a un editor crear, clasificar y publicar, entre otras funcionalidades, cualquier tipo de información en una página web. Generalmente los CMS trabajan contra una base de datos, de modo que el editor simplemente actualiza una base de datos, incluyendo nueva información o editando la existente (Álvarez, 2008).

1.3.1. Joomla

Joomla es un gestor de contenidos para desarrollos web. Se basa en estándares propios de desarrollos web como XHTML (del inglés *eXtensible HyperText Markup Language*), CSS (siglas en inglés de *cascading style sheets*), *JavaScript* y PHP (en inglés Hypertext Preprocessor). *Joomla* es desarrollado por una comunidad de usuarios que lo mejoran, actualizan y lo ponen a disposición para su uso gratuitamente. Permite publicar sitios web cuyos contenidos cambian continuamente, fáciles de administrar y ampliar sin necesidad de realizar una inversión económica en *software*. Permite tanto la publicación por parte del gestor de la página web como la interacción de los usuarios de la web a través de publicación de artículos, encuestas o colaboraciones en la gestión del sitio (Rancel, 2010).

1.3.2. WordPress

WordPress es una plataforma semántica avanzada de publicación personal orientada a la estética, los estándares web y la usabilidad. En *WordPress* se puede escribir, modificar artículos y crear una página web o un blog. Se usa para empezar tu propio sitio o publicar tu propio blog en Internet. Es una aplicación de libre distribución bajo la *General Public License* (GPL). Se puede utilizar y modificar de forma gratuita (Domínguez, 2007).

1.3.3. Drupal

Drupal es un CMS que se utiliza para crear sitios web dinámicos y con diferentes funcionalidades. Cuenta con una amplia y activa comunidad de usuarios y desarrolladores que colaboran conjuntamente en su mejora y ampliación. Esta ampliación es posible gracias a que se trata de un sistema modular con una

arquitectura muy consistente, que permite que los módulos creados por cualquier desarrollador puedan interactuar con el núcleo del sistema y con los módulos creados por otros miembros de la comunidad. Con Drupal es posible implementar diversos sitios web: un blog personal o profesional, un portal corporativo, una tienda virtual, una red social o comunidad virtual (Buitrago, 2011).

Selección del CMS

Se decide utilizar el CMS Drupal en su versión 7.42 debido a las facilidades que brinda para el desarrollo de un portal. Además, posee un alto rendimiento, escalabilidad, un buen gestor de contenido, sin dejar de mencionar el tema de la seguridad, ya que las comunidades de desarrollo que posee están en constante evolución y detectan los problemas de seguridad que puedan aparecer en las versiones del CMS.

1.4. Sistemas Gestores de Bases de Datos

Un Sistema Gestor de Bases de Datos es un conjunto de programas no visibles al usuario final que se encargan de la privacidad, la integridad, la seguridad de los datos y la interacción con el sistema operativo. Proporciona una interfaz entre los datos, los programas que los manejan y los usuarios finales. Cualquier operación que el usuario hace consultando la base de datos está controlada por el gestor. El gestor almacena una descripción de datos en lo que llamamos diccionario de datos, así como los usuarios permitidos y los permisos (Alvarez, 2007).

1.4.1. MySQL

MySQL es un sistema de gestión de bases de datos relacional, licenciado bajo la GPL de la GNU (acrónimo Gnu Not Unix). Su diseño multi-hilo le permite soportar una gran carga de forma muy eficiente. Como base de datos relacional, utiliza múltiples tablas para almacenar y organizar la información. Escrito en los lenguajes de programación C y C++ y se destaca por su gran adaptación a diferentes entornos de desarrollo, permitiendo su interacción con los lenguajes de programación más utilizados como PHP, Perl y Java y su integración en distintos sistemas operativos (Kemppainen, 2015).

1.4.2. PostgreSQL

PostgreSQL es un potente sistema de base de datos objeto-relacional de código abierto. Soporta almacenamiento de objetos binarios grandes, como imágenes, sonidos o vídeo. Cuenta con interfaces nativas de programación para C / C + +, Java, Net, Perl, Python, Ruby, Tcl (del inglés *Tool Command*

Language), ODBC (siglas en inglés *Open DataBase Connectivity*) entre otros. Es altamente escalable, tanto en la enorme cantidad de datos que puede manejar y en el número de usuarios concurrentes que puede administrar. Incluye una biblioteca de funciones estándar con cientos de funciones integradas que van desde las operaciones matemáticas básicas, operaciones con *strings* para criptografía y compatibilidad con Oracle. Los disparadores (*triggers*) y procedimientos almacenados pueden ser escritos en C y se cargan en la base de datos como una biblioteca, lo cual permite una gran flexibilidad y ampliación de sus capacidades. Incluye un *framework* que permite a los desarrolladores definir y crear sus propios tipos de datos personalizados (PostgreSQL, 2010).

Selección del gestor de bases de datos

Teniendo en cuenta las características analizadas de cada gestor de base de datos se decide seleccionar PostgreSQL. A pesar de su gran rapidez y facilidad de uso, MySQL carece de soporte para transacciones, *rollback* y subconsultas, no es viable su uso con grandes bases de datos a las que se acceda continuamente, ya que no implementa una buena escalabilidad. No maneja la integridad referencial, lo cual proporciona una solución pobre para muchos campos de aplicación.

PostgreSQL por su parte posee una gran escalabilidad, ajustarse al número de CPU (del inglés *central processing unit*) y a la cantidad de memoria que posee el sistema de forma óptima, haciéndole ser capaz de soportar una mayor cantidad de peticiones simultáneas de manera correcta. Implementa el uso de *rollback*, subconsultas y transacciones, haciendo su funcionamiento eficaz y ofreciendo soluciones en campos en las que MySQL no podría. Presenta la capacidad de comprobar la integridad referencial, así como también la de almacenar procedimientos en la propia base de datos, equiparándolo con los gestores de bases de datos de alto nivel, como puede ser Oracle.

1.5. Servidores web

Un servidor web es un programa que se ejecuta continuamente en un computador, manteniéndose a la espera de peticiones de ejecución que le hará un cliente o un usuario de Internet. Se encarga de contestar a estas peticiones de forma adecuada, entregando como resultado una página web o información de todo tipo de acuerdo a los comandos solicitados (Morales, 2015).

1.5.1. Apache

Apache Web Server es el servidor web o servidor HTTP (del inglés *Hypertext Transfer Protocol*) con mayor participación en el mundo. Es un sistema de código abierto multiplataforma. Su uso principal es responder consultas HTTP realizadas por navegadores web. Apache responde a la petición de recursos web tanto estáticos como dinámicos y puede trabajar con bases de datos y lenguajes de programación. Es multiplataforma, lo que puede ser ejecutado en diferentes ambientes como Linux, *Windows*, Solaris y Unix. Es altamente configurable y modular, por lo que se puede adaptar para satisfacer diferentes necesidades. Registra en archivos *log*, los accesos que tiene por parte de los usuarios para facilitar la obtención de estadísticas que son usadas para la toma de decisiones por parte del administrador (Uribe, Montoya y Rodríguez, 2015).

1.6. Lenguajes de desarrollo

Los lenguajes de programación son un conjunto de programas que controlan el comportamiento físico y lógico de una máquina, para expresar algoritmos con precisión, o como modo de comunicación humana. Están formados por un conjunto de símbolos y reglas sintácticas y semánticas que definen su estructura y el significado de sus elementos y expresiones (Adamo, 2011).

1.6.1. Pre-Procesador de Hipertexto (PHP 5)

Es un lenguaje de programación interpretado que se utiliza para la generación de páginas web de forma dinámica. Éste código se ejecuta al lado del servidor y se incrusta dentro del código HTML. Puede emplearse en todos los sistemas operativos principales, incluyendo Linux en muchas variantes de Unix (incluyendo HP-UX, Solaris y OpenBSD), *Microsoft Windows*, Mac OS X, RISC OS, entre otros más. PHP admite la mayoría de servidores web de hoy en día, incluyendo Apache, IIS y muchos otros. Esto incluye cualquier servidor web que pueda utilizar el binario de PHP FastCGI, como *lighttpd* y *nginx*, de modo que se tiene la libertad de elegir el sistema operativo y el servidor web.

En este sentido, se tiene la posibilidad de utilizar programación por procedimientos o programación orientada a objetos (POO), o una mezcla de ambas. Con PHP no se está limitado a generar HTML. Entre las capacidades de PHP se incluyen la creación de imágenes, ficheros extensión *.pdf* e incluso películas flash. También se puede generar fácilmente cualquier tipo de texto, como XHTML y cualquier otro tipo de

fichero XML (siglas en inglés de *eXtensible Markup Language*), puede autogenerar estos ficheros y guardarlos en el sistema de ficheros en vez de imprimirlos en pantalla, creando una caché en el lado del servidor para contenido dinámico (Cowburn, 2014).

Según (Wang, 2015) algunas ventajas de PHP son:

- PHP puede realizar páginas dinámicas más rápido que Perl.
- En comparación con otros lenguajes de programación, los procedimientos están incrustados en el documento HTML que ejecuta PHP, por lo que la eficiencia es mucho mayor que en otros lenguajes de programación.
- Soporta las bases de datos y sistemas operativos más utilizados.
- PHP puede utilizar C, C ++ para extender los procedimientos.
- PHP ofrece todas las funciones básicas como lenguaje de codificación.

1.6.2. Lenguaje de Marcas de Hipertexto (HTML 5)

HTML 5 es un estándar universalmente aceptado por la industria. A diferencia de las anteriores actualizaciones HTML, las especificaciones de HTML 5 tienen una vista más amplia de lo que se necesita para mejorar el desarrollo y la programación de diseños web. El principal criterio de diseño de HTML 5 ha sido el de resolver problemas prácticos, lo que hace que se hayan adoptado soluciones orientadas que facilita el trabajo en situaciones reales. Simplifica el trabajo con las nuevas herramientas de gestión de contenidos y facilita la inclusión de elementos multimedia. Con HTML 5 será posible construir páginas de una cierta complejidad y permitirá profundizar en formas más sofisticadas de construir contenidos para Internet (Playán, 2014).

1.6.3. Hojas de estilo en cascada (CSS 3)

Es un lenguaje de hojas de estilos creado para controlar el aspecto o presentación de los documentos electrónicos definidos con HTML y XHTML. CSS es la mejor forma de separar los contenidos y su presentación y es imprescindible para crear páginas web complejas. Mejora la accesibilidad del documento, reduce la complejidad de su mantenimiento y permite visualizar el mismo documento en infinidad de dispositivos diferentes (Eguiluz, 2007).

1.6.4. JavaScript

JavaScript es lo que se conoce como lenguaje *script*, es decir, se trata de un código de programación que se inserta dentro de un documento. Desarrollado por la empresa *Netscape* con la idea de potenciar la creación de páginas web dinámicas para su navegador. Es totalmente distinto de Java. *JavaScript* es más sencillo porque permite insertar código especial dentro del HTML de una página, su función es ampliar las posibilidades de HTML. No crea programas independientes, dependen por completo del código HTML de la página. Es interpretado directamente por el navegador; de hecho, el código *JavaScript* se incrusta dentro del código HTML de la página (Sánchez, 2003).

1.7. Herramientas utilizadas en las pruebas

El proceso de prueba es repetitivo y necesita de la atención del probador en todo momento. Desde hace un tiempo se comenzaron a automatizar estas pruebas a través de herramientas que facilitan, agilizan y aseguran este complejo proceso. A continuación se muestran algunas de estas herramientas:

1.7.1. Apache Jmeter

La aplicación Jmeter es un *software* de código abierto, creada en java y diseñada para probar el comportamiento funcional de un sitio web y medir el rendimiento. Fue diseñado originalmente para pruebas de aplicaciones web, pero desde entonces se ha expandido a otras funciones de prueba. Jmeter puede ser utilizado para probar el rendimiento en recursos estáticos y dinámicos, lenguajes dinámicos como PHP, Java, ASP.NET, objetos Java, bases de datos y consultas. Se puede utilizar para simular una carga pesada en un servidor, grupo de servidores, la red o el objeto para probar su resistencia o para analizar el rendimiento general bajo diferentes tipos de carga (Apache JMeter, 2015).

1.7.2. Acunetix Web Vulnerability Scanner 8

Acunetix Web Vulnerability Scanner es una herramienta para realizar pruebas de seguridad en aplicaciones web. Acunetix ha sido pionera en la aplicación web de tecnología de análisis de seguridad. Sus ingenieros se han centrado en la seguridad web desde 1997 y ha desarrollado una ingeniería líder en el análisis de sitios web y detección de vulnerabilidades (Acunetix, 2016).

Algunas de sus características son:

- Las herramientas de testeo de inyección SQL y de *crosssite scripting* más avanzadas y profundas de la industria.
- Herramientas avanzadas de penetración, como HTTP Editor y HTTP Fuzzer.
- Herramientas para fácil aseguramiento de formularios web y contraseñas.
- Soporte para páginas con *captcha*, *single sign-on* y mecanismos con factor de autenticación.
- Facilidad de generación de informes amplios, incluyendo informes de cumplimiento PCI.
- El escaneo inteligente detecta el tipo de servidor web y lenguaje de la aplicación.
- Acunetix escanea y analiza sitios web incluyendo contenido flash, SOAP y AJAX.
- Permite explorar un servidor web y ejecutar comprobaciones de seguridad contra los servicios de red que se ejecutan en el servidor.

1.8. Entorno de desarrollo integrado

Un entorno de desarrollo integrado (en inglés *Integrated Development Environment* o IDE) es un programa compuesto por una serie de herramientas que utilizan los programadores para desarrollar código. Esta herramienta puede estar presentada por su utilización con un único lenguaje de programación o bien puede dar cavidad a varios de estos (Pedre, 2012).

1.8.1. Netbeans

Netbeans es un entorno de desarrollo gratuito y de código abierto. Permite el uso de un amplio rango de tecnologías de desarrollo tanto para escritorio, como aplicaciones Web, o para dispositivos móviles. Da soporte a tecnologías como: Java, PHP, Groovy, C/C++, HTML5, entre otros. Además puede instalarse en varios sistemas operativos: *Windows*, Linux, Mac OS (Genbeta Dev, 2014).

1.9. Herramientas CASE

CASE (del inglés *Computer-Aided Systems Engineering*) cuyo significado en español es Ingeniería del *Software* Asistida por Computadora, es la aplicación de tecnología informática a las actividades, las técnicas y las metodologías propias del desarrollo de sistemas. Su objetivo es automatizar o apoyar una o más fases del ciclo de vida del desarrollo de sistemas (Martínez, 2007).

1.9.1. Visual Paradigm

Visual Paradigm For UML (por sus siglas en inglés *Unified Modeling Language*) es una herramienta Case. Provee el modelado de procesos de negocios, además de un generador de mapeo de objetos-relacionales para los lenguajes de programación Java .NET y PHP. Integra herramientas Java como: Eclipse/IBM WebSphere JBuilder NetBeans IDE Oracle JDeveloper BEA Weblogic. Es un producto galardonado que facilita a las organizaciones, la diagramación visual y el diseño de sus proyectos de sistema, que les brinda la posibilidad de integrar y desplegar sus aplicaciones empresariales de misión crítica y de sus bases de datos subyacentes (Sierra, 2007).

1.10. Metodologías de desarrollo de software

Una metodología de desarrollo de *software* describe un entorno que se utiliza para organizar, planificar y dirigir el proceso de desarrollo de *software*. Existen muchas metodologías de desarrollo de *software* y contienen algunas etapas básicas del ciclo de vida del desarrollo de *software*. Estas etapas son: planificación, análisis, diseño, implementación y mantenimiento (Verma, Pandey y Bansal, 2014).

1.10.1. Proceso Unificado Ágil (AUP)

El Proceso Unificado Ágil (AUP del inglés *Agile Unified Process*) es una versión simplificada del Proceso Unificado Racional (RUP por sus siglas en inglés *Rational Unified Process*). 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. El AUP aplica técnicas ágiles incluyendo:

- Desarrollo dirigido por pruebas (*test driven development* - TDD en inglés).
- Modelado ágil.
- Gestión de cambios ágiles.
- 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.

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 el mismo.

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 completamente en el ambiente de desarrollo.
4. Transición: el sistema se lleva a los entornos de pre-producción donde se somete a pruebas de validación y aceptación y finalmente se despliega en los sistemas de producción (Sánchez, 2015).

Conclusiones parciales

En la evaluación de las tendencias de portales web similares a la propuesta de solución presentada en este trabajo se observan aspectos generales como mala navegación, los recursos no tienen un autor definido y en algunos casos no presentan buscadores alojados en sus portales, sin embargo, para ellos resultan viables. Teniendo en cuenta los aspectos negativos encontrados, no se consideran posibles soluciones para el nuevo portal web. Aun así, fue posible identificar a partir de ellos, nuevos elementos que tributan al perfeccionamiento de la propuesta de solución.

Con la selección de la metodología, herramientas y tecnologías con soporte multiplataforma y basadas en *software* libre, se logró obtener una base tecnológica adecuada que permitirá el desarrollo de la solución.

Capítulo 2: Caracterización del portal web del Centro de Información Científico-Técnica de la UCI

En este capítulo se explica cómo se desarrolla el flujo actual de los procesos y se describe la propuesta de solución para resolver el problema planteado. Por otra parte, se especifican los requisitos funcionales y no funcionales y los elementos fundamentales del diseño y de la arquitectura que se deben tener en cuenta para el desarrollo del sistema, ayudado por la realización del modelado de diagramas.

2.1 Propuesta del sistema

Para dar cumplimiento al objetivo planteado y teniendo en cuenta las necesidades de mejorar el portal que se encuentra actualmente en línea en el CICT de la Universidad, se propone la implementación de un portal web que permita perfeccionar la gestión de los recursos de información que se han automatizado en el Centro de Información Científico-Técnica de la Universidad de las Ciencias Informáticas, de forma tal que los usuarios puedan consultar dicha información. En la solución existirá un espacio para mostrar las nuevas adquisiciones que ingresan en el CICT y las importantes noticias enmarcadas en el mundo del conocimiento y la información. Otra de las acciones a ofrecer en el sistema es el control de visitas al portal web, el cual permite obtener la cantidad de visitas recibidas en varios instantes de tiempo.

Otro de los aspectos a considerar en la solución propuesta es el acceso a sitios de interés publicados actualmente en Internet, lo que posibilita que se pueda acceder a determinadas fuentes externas de información. También debe brindar acceso a diferentes sistemas o servicios, como es el caso del repositorio institucional y el servicio de vigilancia tecnológica, ambos desarrollados por terceros. La aplicación permitirá al personal del CICT encargado de realizar la actualización del portal, hacerlo de una forma rápida y fácil. La solución que se propone proporciona que la información contenida en el portal se encuentre de forma organizada y con una correcta estructura.

2.2 Modelado conceptual

Los modelos conceptuales, se entienden en ingeniería del *software* como un tipo de modelos relativamente poco sofisticados y, por lo tanto, más simples de comprender. Para lograr favorecer la comprensión de las necesidades del usuario y los requisitos de *software* de la propuesta de solución se construye el siguiente modelo conceptual.

Descripción de los elementos del modelo conceptual

Sitio web: sistema que mostrará todos los contenidos de la biblioteca.

Contenido: hace referencia a los diferentes contenidos del portal, estos son: Adquisiciones, Catálogos, Servicio, Noticia, Publicaciones, Base de datos y Eventos.

Noticia: hace referencia al tipo de contenido de la sección “Noticia y Evento”.

Evento: hace referencia al tipo de contenido de la sección “Noticia y Evento”.

Adquisiciones: hace referencia al tipo de contenido de la sección “Nuevas adquisiciones”.

Servicio: hace referencia al tipo de contenido de la sección “Servicio”.

Publicaciones: hace referencia al tipo de contenido de la sección “Publicaciones”.

Catálogos externos: hace referencia al tipo de contenido de la sección “Catálogos externos”.

Base de datos: hace referencia al tipo de contenido de la sección “Base de datos”.

Contacto: permite a los usuarios contactar con los especialistas del CICT.

Usuario: persona que interactúa con el portal.

Administrador: usuario encargado de velar por la veracidad y autenticidad de los materiales publicados.

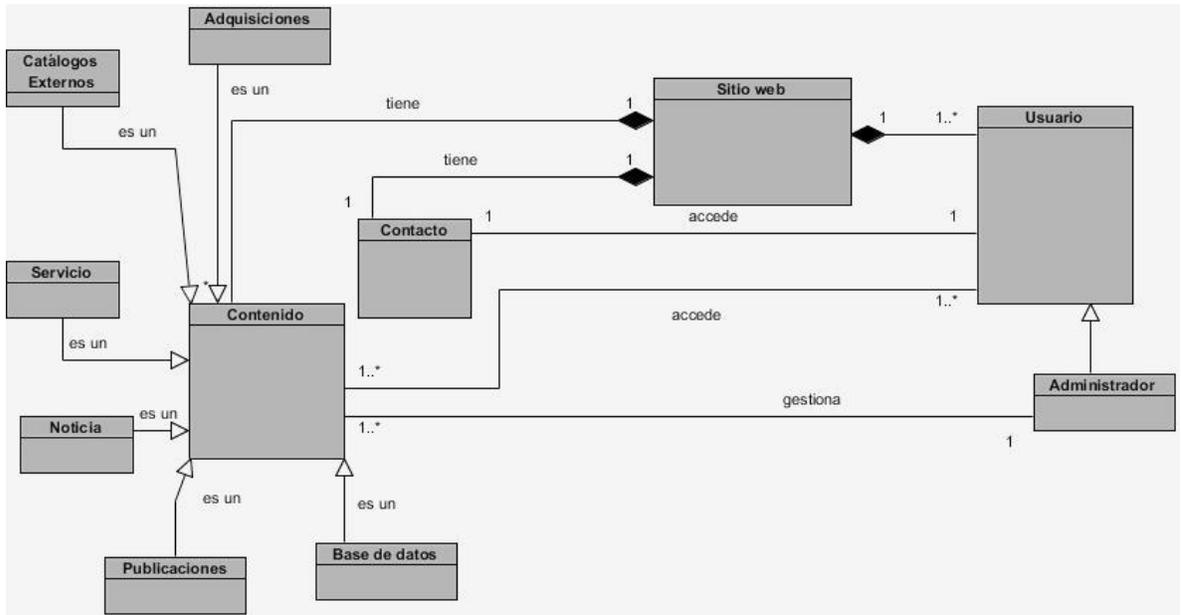


Ilustración 1: Modelo conceptual de la propuesta de solución

2.3 Levantamiento de requisitos

Los requisitos definen lo que el sistema debe hacer, para lo cual se identifican las funcionalidades requeridas y las restricciones que se imponen. Los requisitos funcionales (RF) son las condiciones que el sistema debe cumplir, mientras que los requisitos no funcionales (RNF) son las características que el producto debe tener.

2.3.1 Requisitos Funcionales

Tabla 2: Requisitos funcionales

No	Nombre	Descripción	Complejidad
RF1	Autenticar usuario	El usuario introduce sus datos, el sistema lo comprueba y si son válidos, el	Media

		usuario queda autenticado con el nivel de privilegios asignados, si los datos no son válidos el sistema muestra un mensaje de error.	
RF2	Crear usuario	El sistema permite que los usuarios con rol administrador creen los usuarios.	Media
RF3	Editar usuario	El sistema permite que los usuarios con rol administrador editen los usuarios.	Baja
RF4	Eliminar usuario	El sistema permite que los usuarios con rol administrador eliminen cuentas de usuarios existentes.	Alta
RF5	Listar usuario	El sistema permite que los usuarios con rol administrador visualicen el listado de usuarios existentes.	Baja
RF6	Mostrar usuario	El sistema permite que los usuarios con rol administrador visualicen los datos de los usuarios existentes.	Alta
RF7	Crear adquisición	El sistema permite que los usuarios con rol de editor creen las adquisiciones.	Alta
RF8	Editar adquisición	El sistema permite que los	Media

		usuarios con rol de editor editen las adquisiciones.	
RF9	Mostrar adquisición	El sistema permite que todos los usuarios observen las adquisiciones.	Media
RF10	Eliminar adquisición	El sistema permite que los usuarios con rol de editor eliminen las adquisiciones.	Baja
RF11	Crear noticia	El sistema permite que los usuarios con rol de editor creen las noticias.	Alta
RF12	Editar noticia	El sistema permite que los usuarios con rol de editor editen las noticias.	Media
RF13	Mostrar noticia	El sistema permite que todos los usuarios observen las noticias.	Media
RF14	Imprimir noticia	El sistema permite a los usuarios imprimir las noticias	Media
RF15	Exportar Noticia	El sistema permite exportar noticias a formato pdf.	Media
RF16	Eliminar noticia	El sistema permite que los usuarios con rol de editor eliminen las noticias.	Baja
RF17	Mostrar información de estadísticas de visita	El sistema permite que todos los usuarios observen las estadísticas de visitas	Baja

		realizadas.	
RF18	Búsqueda simple de contenido	El sistema permite la búsqueda simple de la información que contiene.	Media
RF19	Búsqueda avanzada de contenido	El sistema permite la búsqueda avanzada de la información que contiene.	Alta
RF20	Mostrar mapa de sitio	El sistema permite que todos los usuarios observen el mapa de sitio.	Alta
RF21	Crear producto y servicio	El sistema permite que los usuarios con rol de administrador creen los productos y servicios.	Alta
RF22	Editar producto y servicio	El sistema permite que los usuarios con rol de administrador editen los productos y servicios.	Media
RF23	Mostrar producto y servicio	El sistema permite que todos los usuarios observen los productos y servicios.	Media
RF24	Eliminar producto y servicio	El sistema permite que los usuarios con rol de administrador eliminen los productos y servicios.	Baja
RF25	Insertar enlaces a las bases de datos	El sistema permite que los usuarios con rol de	Alta

		administrador inserten enlaces a las bases de datos.	
RF26	Editar enlaces a las bases de datos	El sistema permite que los usuarios con rol de administrador editen los enlaces a las bases de datos.	Media
RF27	Mostrar enlaces a las bases de datos	El sistema permite que todos los usuarios observen los enlaces a las bases de datos.	Media
RF28	Eliminar enlaces a las bases de datos	El sistema permite que los usuarios con rol de administrador eliminen los enlaces a las bases de datos.	Baja
RF29	Insertar enlaces a las publicaciones seriadas	El sistema permite que los usuarios con rol de administrador inserten los enlaces a las publicaciones seriadas.	Alta
RF30	Editar enlaces a las publicaciones seriadas	El sistema permite que los usuarios con rol de administrador editen los enlaces a las publicaciones seriadas.	Media
RF31	Mostrar enlaces a las publicaciones seriadas	El sistema permite que todos los usuarios observen los enlaces a las publicaciones seriadas.	Media
RF32	Eliminar enlaces a	El sistema permite que los	Baja

	las publicaciones seriadas	usuarios con rol de administrador eliminen los enlaces a las publicaciones seriadas.	
RF33	Crear aviso	El sistema permite que los usuarios con rol de administrador creen avisos.	Alta
RF34	Editar aviso	El sistema permite que los usuarios con rol de administrador editen los avisos.	Media
RF35	Mostrar aviso	El sistema permite que todos los usuarios visualicen los avisos.	Media
RF36	Eliminar aviso	El sistema permite que los usuarios con rol de administrador eliminen los avisos.	Baja
RF37	Insertar boletín	El sistema permite que los usuarios administrador inserten un boletín con el contenido que desea enviar a los <i>suscriptores</i> del sitio.	Alta
RF38	Suscribir un usuario al Boletín	El sistema permite que los usuarios autenticados o anónimos se suscriban al boletín.	Alta
RF39	Enviar boletín	El sistema permite que los	Alta

		usuarios administrador envíen un boletín a los <i>suscriptores</i> .	
RF40	Editar Boletín	El sistema permite que los usuarios con rol administrador editen los boletines existentes.	Media
RF41	Mostrar Boletín	El sistema permite que los usuarios autenticados o anónimos visualicen el boletín.	Media
RF42	Enviar confirmación de la suscripción al boletín	El sistema permite enviar un correo de confirmación al usuario que se suscribe.	Media
RF43	Mostrar lista de usuarios suscritos a un boletín	El sistema permite que los usuarios con rol de administrador visualicen el listado de los usuarios suscritos a un boletín.	Baja
RF44	Dar baja a un usuario del Boletín	El sistema permite que los usuarios autenticados o anónimos se den baja del boletín.	Baja
RF45	Eliminar boletín	El sistema permite que los usuarios administrador eliminar un boletín.	Baja
RF46	Crear vínculos bibliotecarios	El sistema permite que los usuarios con rol de administrador creen enlaces a otros sitios bibliotecarios.	Alta

RF47	Editar vínculos bibliotecarios	El sistema permite que los usuarios con rol de administrador editen los enlaces a sitios bibliotecarios.	Media
RF48	Mostrar vínculos bibliotecarios	El sistema permite que todos los usuarios visualicen los enlaces a sitios bibliotecarios.	Media
RF49	Eliminar vínculos bibliotecarios	El sistema permite que los usuarios con rol de administrador eliminen los enlaces a sitios bibliotecarios.	Baja
RF50	Mostrar RSS	El sistema permite que los usuarios autenticados o anónimos soliciten RSS.	Alta
RF51	Cambiar contraseña	El sistema permite que los usuarios registrados en el mismo cambian su contraseña.	Media
RF52	Listar contenido	El sistema permite que los usuarios con rol de administrador visualicen el listado de contenidos existentes.	Baja
RF53	Filtrar contenido	El sistema permite que los usuarios con rol de administrador filtren los contenidos mediante criterios de búsquedas definidos por el	Alta

		propio usuario.	
RF54	Editar página básica	El sistema permite a los usuarios con rol editor editen los contenidos de tipo página básica.	Baja
RF55	Mostrar página básica	El sistema permite que los usuarios autenticados o anónimos visualicen las páginas básicas.	Alta

2.3.2 Requisitos no Funcionales

- RNF1.** El sistema mostrará un mapa del sitio para proporcionarle al usuario facilidad de navegación dentro del sistema.
- RNF2.** Diseño limpio y claro que facilite la navegación.
- RNF3.** Se debe lograr una estructura simple y lo más intuitiva posible de la arquitectura de la información.
- RNF4.** Se asignarán los permisos de acceso, escritura, lectura en dependencia del rol que desempeñe cada usuario del sistema.
- RNF5.** El sistema debe permitir la creación de usuarios con diferentes niveles de acceso.
- RNF6.** El sistema debe poseer un alto grado de escalabilidad, para permitir incorporarle nuevas funcionalidades.
- RNF7.** Lenguaje de programación PHP5.
- RNF8.** Lenguaje de marcas de hipertextos en su versión 5.
- RNF9.** Hojas de estilo en cascada en su versión 3.

- RNF10.** Entorno de desarrollo integrado NetBeans versión 8.
- RNF11.** Sistema de gestión de contenido Drupal versión 7.42.
- RNF12.** Por parte del cliente se requiere un navegador capaz de interpretar *JavaScript*.
- RNF13.** Se podrá acceder a la aplicación desde cualquier navegador web, ya sea Internet Explorer a partir de la versión 7, Mozilla Firefox, Opera, Chrome y Safari.
- RNF14.** El sistema debe poseer un diseño web adaptable a diferentes resoluciones.
- RNF15.** Uso de la licencia Apache *Software*.
- RNF16.** Uso de licencia GNU/GPL para CMS Drupal.
- RNF17.** Uso de licencia PHP.
- RNF18.** Uso de la licencia BSD de PostgreSQL.
- RNF19.** Servidor Dual Core 2.10 GHz, 4GB RAM, 200 GB de espacio libre en disco.

2.4 Historias de Usuario

Las historias de usuarios son la forma en que se especifican los requisitos del sistema, según la metodología de desarrollo utilizada. El tratamiento de las historias de usuario es muy dinámico y flexible, en cualquier momento estas pueden romperse, remplazarse por otras más específicas o generales, añadirse nuevas o ser modificadas. En la claridad de su descripción se encuentra el éxito del proyecto por lo cual el contenido que ellas abarcan debe ser concreto y sencillo.

Tabla 3: Historia de Usuario. Autenticar usuario

Número: 1	Nombre del requisito: Autenticar usuario
Programador: Oberman Eriza Oviedo	Iteración asignada: Primera iteración

Prioridad: Baja	Tiempo estimado: 1día
Riesgo en desarrollo: Medio	Tiempo real: 1día
<p>Descripción: Los usuarios con rol Superadmin o administrador que se hayan registrado con anterioridad en el sistema se pueden autenticar, para ello deben llenar los siguientes campos:</p> <ul style="list-style-type: none"> • Nombre de usuario (Obligatorio. Campo de texto. Permite todos los caracteres) • Contraseña (Obligatorio. Campo de texto. Permite todos los caracteres) 	
<p>Observaciones:</p> <ol style="list-style-type: none"> 1. Si el usuario introduce la información de forma correcta, el sistema emite un mensaje notificando que se ha autenticado satisfactoriamente el usuario. 2. Si el usuario introduce la información de forma incorrecta, el sistema emite un mensaje notificando el error. 3. Si el usuario introduce la información dejando campos obligatorios vacíos, el sistema emite un mensaje indicándole que los campos obligatorios deben de llenarse. 	

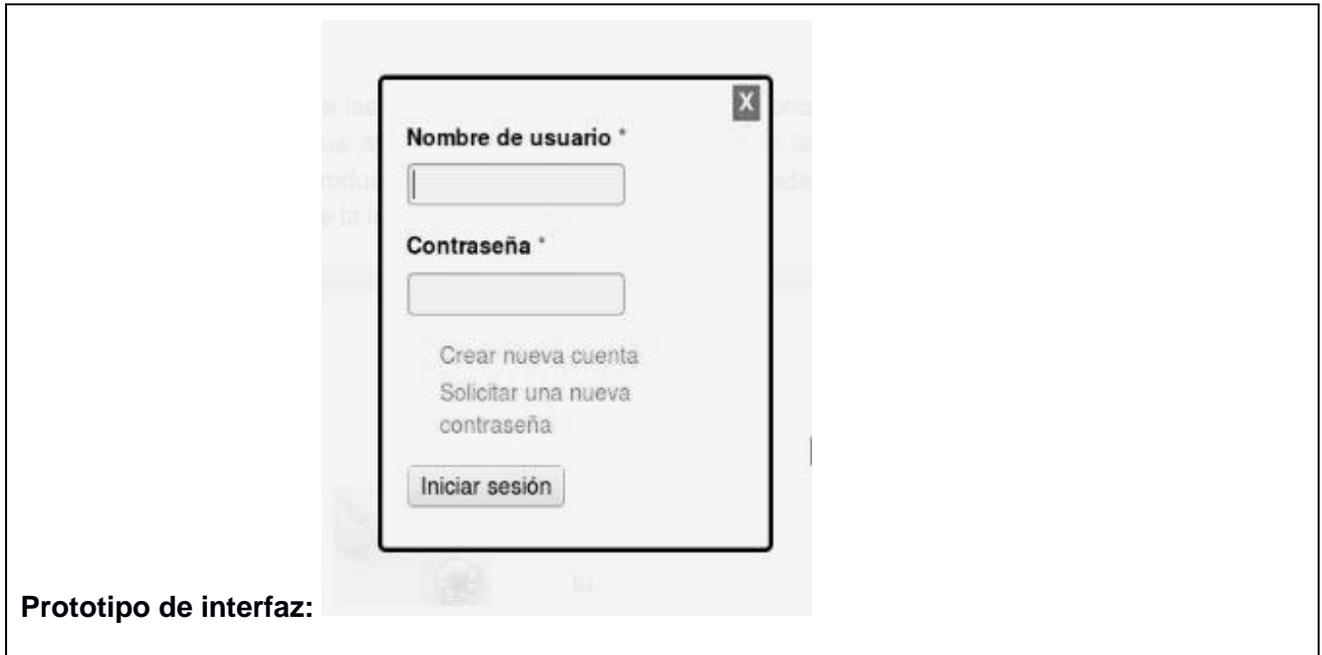


Tabla 4: Historia de Usuario. Crear usuario

Número: 2	Nombre del requisito: Crear usuario
Programador: Oberman Eriza Oviedo	Iteración asignada: Primera iteración
Prioridad: Alta	Tiempo estimado: 1día
Riesgo en desarrollo: Medio	Tiempo real: 1día
<p>Descripción: Los usuarios con rol Superadmin pueden crear nuevos usuarios en el sistema, para ello deben llenar los siguientes campos:</p> <ul style="list-style-type: none"> • Nombre de usuario (Obligatorio. Campo de texto. Long min. 5, máx. 60 caracteres. Permiten los caracteres: (a-z, A-Z, 0-9)). • Dirección de correo electrónico (Obligatorio. Campo de texto. Long máx. 254 caracteres. Permite direcciones de correo estructuralmente válidas (Estructura: 	

[usuario@subdominios.dominio](#)).

- **Contraseña** (Obligatorio. Campo de texto. Long min. 6, máx. 128 caracteres. Permite los caracteres a-z, A-Z, 0-9 y signos de puntuación).
- **Confirmar contraseña** (Obligatorio. Campo de texto. Long min. 6, máx. 128 caracteres. Permite los caracteres a-z, A-Z, 0-9 y signos de puntuación).
- **Estado** (Opcional. Campo de selección. Valores (Bloqueado, Activo). Valor por defecto (activo)).
- **Roles** (Opcional. Campo de selección múltiple. Valores (Superadmin, administrador)).
- **Notificar al usuario acerca de su nueva cuenta** (Opcional. Campo de selección. Valor por defecto (Desmarcado)).

Observaciones:

1. Si el usuario introduce la información de forma correcta, el sistema emite un mensaje notificando que se ha creado satisfactoriamente el usuario.
2. Si el usuario introduce la información de forma incorrecta, el sistema emite un mensaje notificando el error.
3. Si el usuario introduce la información dejando campos obligatorios vacíos, el sistema emite un mensaje indicándole que los campos obligatorios deben de llenarse.
4. Si el usuario ya existe, el sistema emite un mensaje notificando que el usuario ya está en el sistema.

Prototipo de interfaz: No aplica

2.5 Descripción de estilos arquitectónicos y patrones de diseño

Analizando que una arquitectura de *software* es de especial importancia, ya que la manera en que se estructura un sistema tiene un impacto directo sobre la capacidad de este para satisfacer lo que se conoce como los atributos de calidad del sistema y que los patrones de diseño son una guía para resolver problemas comunes en programación se describe la arquitectura de *software* y los patrones de diseños utilizados en el desarrollo de la propuesta de solución.

2.5.1 Estilo arquitectónico

Al utilizarse el CMS Drupal para el desarrollo del portal, la arquitectura y los patrones son heredados del mismo. Drupal no presenta un estilo arquitectónico único por lo que el autor de la presente investigación selecciona la arquitectura n-capas (Ilustración 2²) ya que en lugar de considerar el sitio web como un conjunto de páginas interrelacionadas, Drupal estructura los contenidos en una serie de elementos básicos como: los nodos, módulos, bloques y menús, permisos de usuario y plantillas.

- La primera capa representa la información almacenada en los Nodos (tipos de contenido a publicar ejemplo noticias, avisos, adquisiciones).
- En la segunda se encuentran los Módulos que son utilizados en la aplicación (ejemplos los módulos *ctool*, *views*, *jcarousel*), que son los elementos que operan sobre los Nodos y otorgan funcionalidad a Drupal permitiendo incrementar sus capacidades o adaptarlas a las necesidades de cada sitio web.
- La tercera esta descrita por los Bloques y Menús que permiten estructurar y organizar los contenidos en la página web (ejemplo: los menús *¿Quiénes Somos?*, *Enlaces* y bloque de adquisiciones, *Video*).
- La cuarta recoge los Permisos de Usuarios donde Drupal dispone de un registro de usuarios y de roles que permiten especificar qué tareas pueden realizar y a qué contenidos puede acceder cada tipo de usuario (ejemplos: roles de administrador, editor y usuario).
- La quinta y última capa maneja las Plantillas y es la que establece la apariencia gráfica o estilo de la información que se le muestra al usuario (ejemplo: el tema biblioteca).

² Imagen tomada de <http://www.cursosdrupal.com/content/arquitectura>

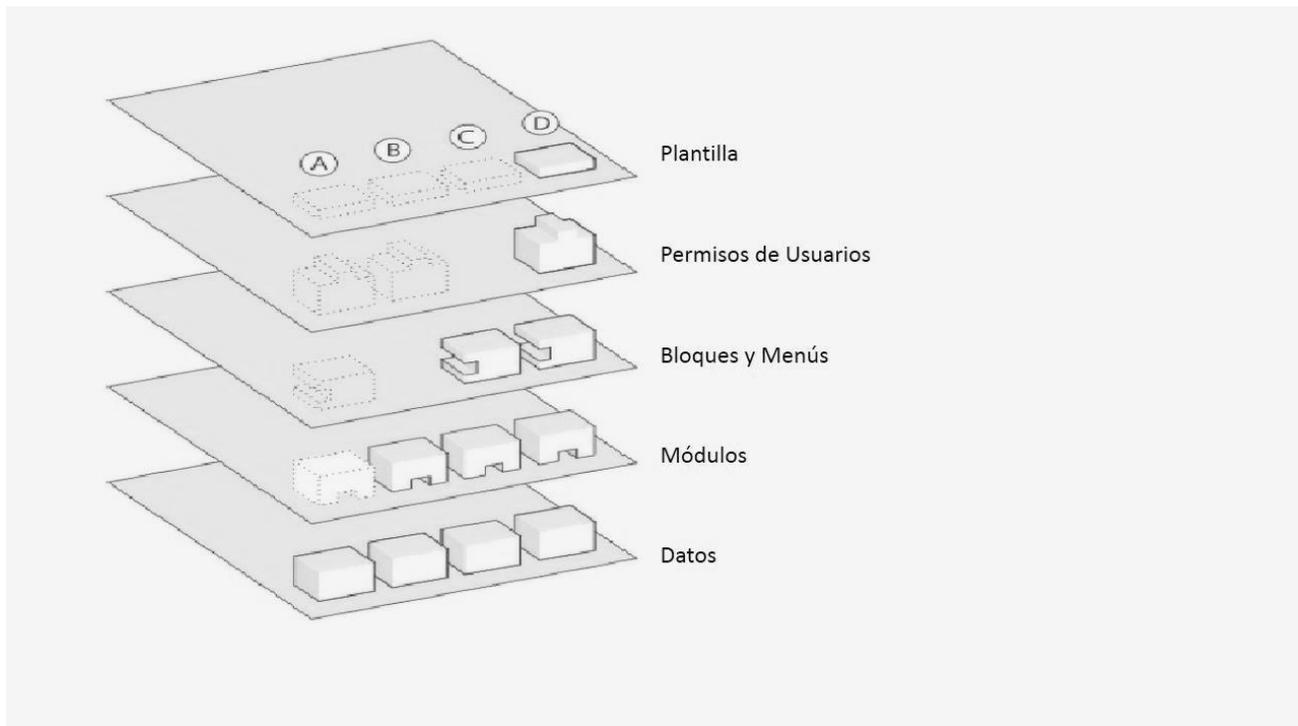


Ilustración 2: Arquitectura de la propuesta de solución.

2.5.2 Patrones de diseño

De los patrones de diseño que utiliza Drupal, el portal emplea los siguientes:

Singleton: este patrón está diseñado para que solo permita que se haga una instancia de un elemento cuando este no existe. Dentro del núcleo de Drupal se utiliza este patrón de diseño pensando en los módulos y temas de Drupal como objetos para llevar a cabo la gestión de dichos elementos, pues Drupal solo crea una instancia de un nodo, cuando este nodo no existe. En el portal puede evidenciarse este patrón cuando el administrador crea un tipo contenido “Noticia”. Primeramente, verifica que no exista en la base de datos y luego lo crea.

Command: Drupal utiliza este patrón para reducir el número de funciones que son necesarias para la aplicación, pasando la operación como un parámetro, junto con los argumentos y así agilizar la ejecución de ciertas tareas en el sistema. En el portal se emplea a la hora de revertir cambios realizados en la forma de representar las imágenes en los tipos de contenido “Noticia”.

Chain of Responsibility: el sistema de menús de Drupal es la evidencia de este patrón, en dependencia de la petición recibida por un módulo, se delega la responsabilidad a la función encargada de atenderla. De esta forma se continúa la cadena hasta que un módulo atienda la petición o la cadena se agote. Un ejemplo en el portal puede verse cuando se selecciona el vínculo “Búsqueda avanzada”. El sistema detecta el módulo de búsqueda que puede encargarse de la petición, verifica si este tiene permisos para usar esta función y entonces permite acceder a su contenido.

2.6 Diagrama de Clases del Diseño

El diagrama de clases es el diagrama principal de diseño y análisis para un sistema. En él, la estructura de clases del sistema se especifica con relaciones entre clases y estructuras de herencia. Durante el análisis del sistema, el diagrama se desarrolla buscando una solución ideal. Durante el diseño se usa el mismo diagrama y se modifica para satisfacer los detalles de las implementaciones.

La representación de los paquetes estará conformada por los módulos de Drupal.

Themes: incluye las distintas plantillas, cuando se desee incluir un nuevo diseño, se copia la plantilla dentro de esta carpeta.

Includes: contiene un conjunto de ficheros indispensables para el correcto funcionamiento del CMS, como por ejemplo la base de datos que provee las funcionalidades de acceso a la base de datos de Drupal.

Modules: comprende todos los módulos que permiten las distintas funcionalidades del CMS; cuando desee incorporar un nuevo módulo solo se tiene que copiar en dicha carpeta.

Scripts: abarca un conjunto de ficheros indispensables para el funcionamiento, fundamentalmente orientados al aspecto visual, CSS y JavaScript.

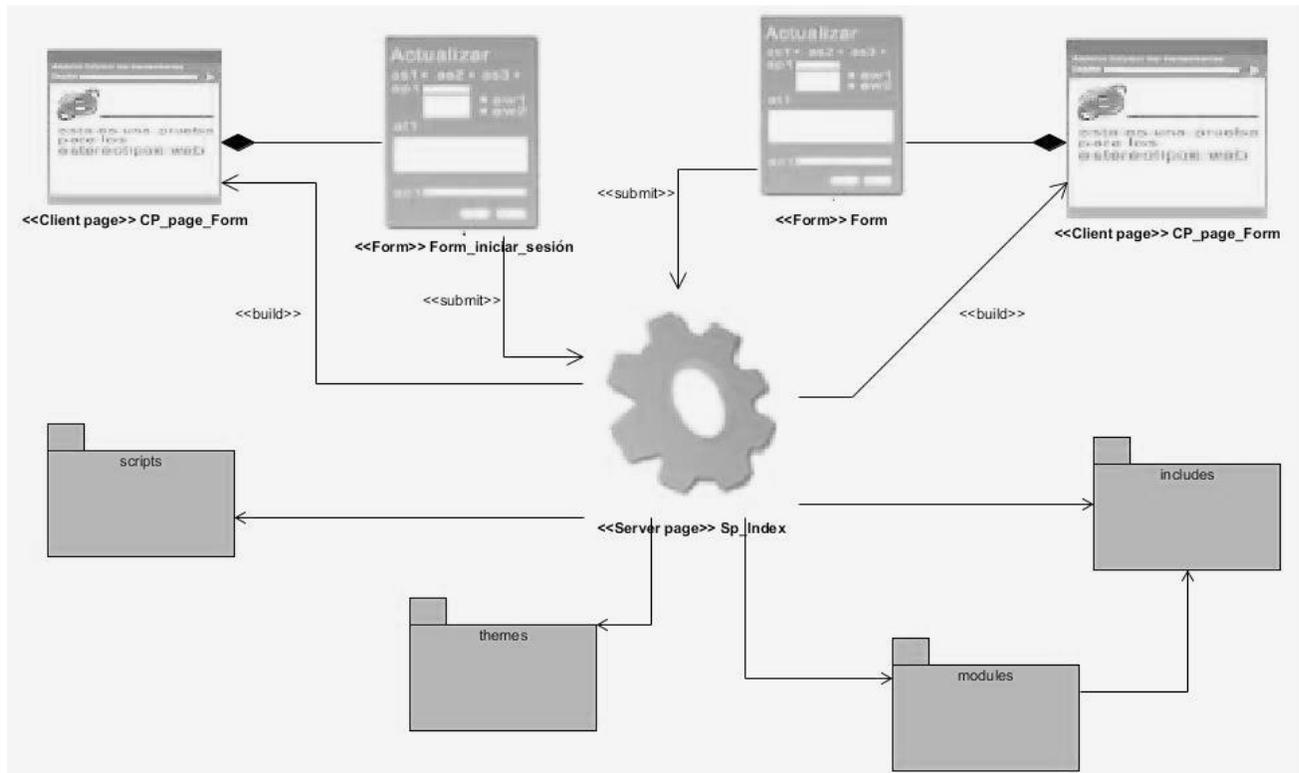


Ilustración 3: Diagrama de clase del diseño

2.7 Modelo de Despliegue

El diagrama de despliegue es un modelo de objetos que describe la distribución física del sistema. Es una colección de nodos y arcos, donde cada nodo representa un recurso de cómputo, normalmente un procesador o un dispositivo de *hardware* similar. Muestra la configuración de los componentes *hardware*, los procesos, los elementos de procesamiento en tiempo de ejecución y los objetos que existen en tiempo de ejecución.

El nodo que representa la PC cliente es un conjunto de computadoras utilizadas por los usuarios para actualizar y visualizar la información que se encuentra en el servidor web a través de un navegador. La comunicación entre las PC clientes y el servidor web se establece utilizando el protocolo de comunicación segura HTTPS. El servidor de base de datos, que representa un servidor PostgreSQL, se encarga de mantener las informaciones generadas como parte de la base de datos del CMS Drupal y permite el acceso a ella mediante el servidor web. Estos dos servidores se comunican mediante la familia de

protocolos TCP (por sus siglas en inglés *Transmission Control Protocol*). El servidor de correo se encarga del envío de los correos del boletín y los formularios de contactos y adquisiciones, todo a través del protocolo SMTP (*Simple Mail Transfer Protocol*). La autenticación al sistema se realiza mediante protocolo LDAP (*Lightweight Directory Access Protocol*) al servidor LDAP.

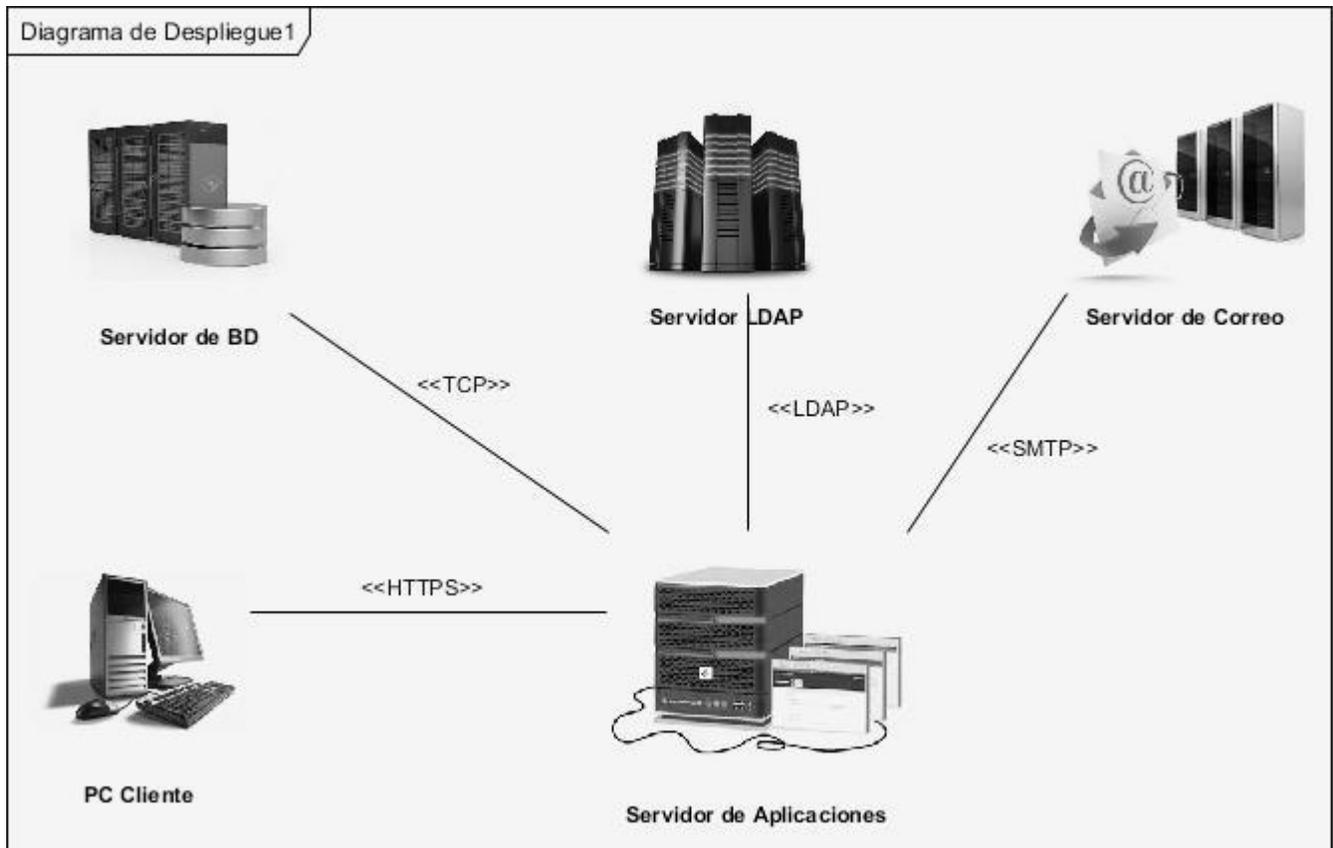


Ilustración 4: Diagrama del modelo de despliegue

Conclusiones parciales

Durante el análisis y diseño del portal web para el CICT, la representación y descripción de los artefactos generados garantizaron un mejor entendimiento del flujo de trabajo. La especificación de los requisitos funcionales y no funcionales del sistema dieron paso a una mejor comprensión, por parte del autor, de los resultados que se pretenden obtener de una manera precisa y sirvieron de guía para la codificación del sistema. La definición de la arquitectura y los patrones de diseño a utilizar, permitieron establecer las

bases para fomentar la reutilización y las buenas prácticas de programación durante la fase de implementación.

Capítulo 3: Implementación y validación del portal web para el Centro de Información Científico-Técnica de la UCI

En este capítulo se exhiben los diagramas asociados, estándares de codificación y diseños de casos de prueba a utilizar en la validación del sistema y se analizan los resultados de las pruebas realizadas que permiten evaluar la calidad de la propuesta de solución.

3.1. Diagrama de componente

El diagrama de componentes establece la relación entre los componentes de *software* (librerías, binarios, ejecutables y códigos fuentes), dependencias, comunicación y ubicación en el sistema.

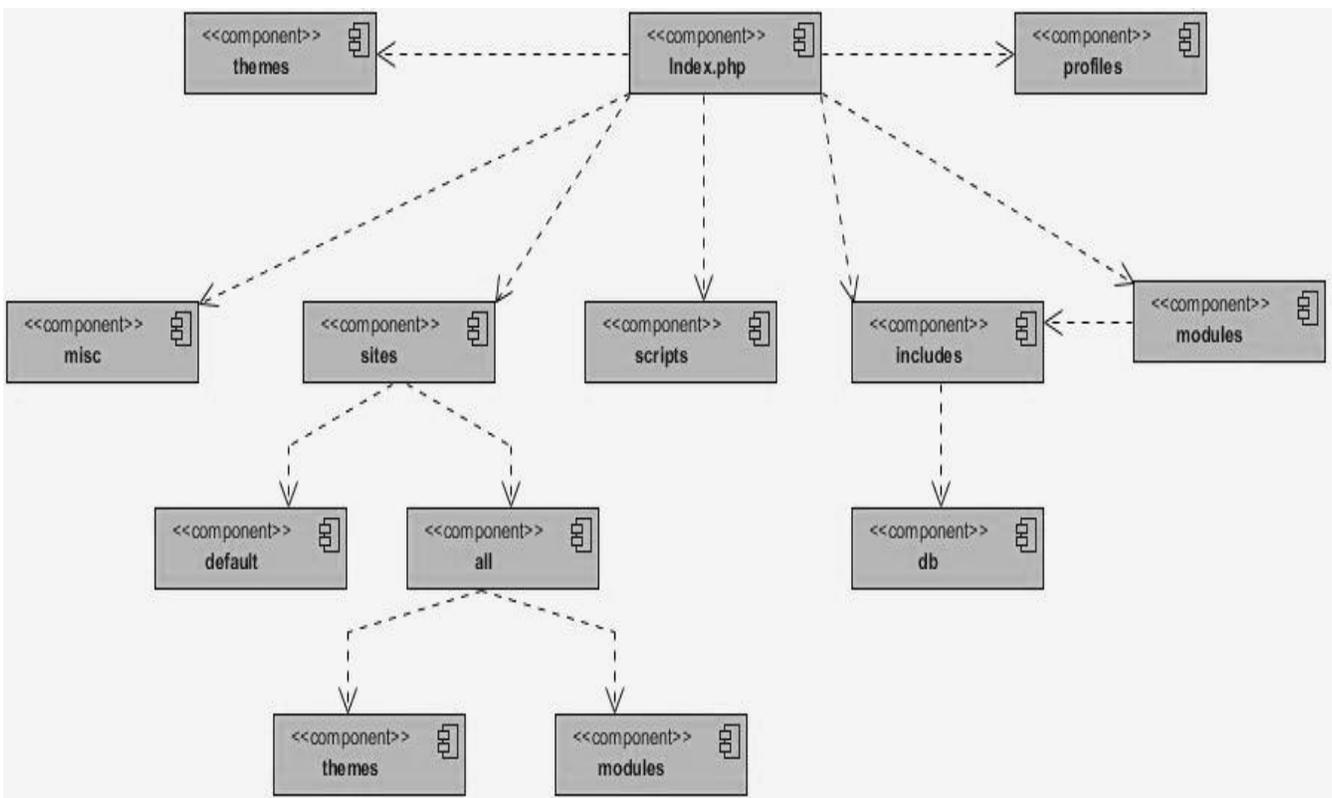


Ilustración 5: Diagrama de componente

A continuación, se describen cada uno de los componentes representados en el diagrama:

- ***index.php***: punto de inicio de la aplicación, a partir de esta entrada se solicitan los diferentes módulos del sistema de gestión de contenidos.
- ***themes***: conjunto de temas que vienen por defecto con la distribución del CMS Drupal.
- ***profiles***: contiene los perfiles de instalación del CMS Drupal.
- ***misc***: este directorio incluye un grupo de archivos necesarios para el sistema como: *JavaScript*, CSS e imágenes.
- ***sites***: se encuentran elementos extras y modificaciones que se añaden al sistema general.
- ***scripts***: Contienen utilidades adicionales que no utiliza Drupal directamente, pero que pueden ser utilizadas desde la línea de comando *Shell*.
- ***includes***: contiene los ficheros indispensables para el funcionamiento de Drupal.
- ***modules***: conjunto de módulos que son parte del núcleo de Drupal y permiten el funcionamiento del sistema.

3.2. Estándares de codificación

Los estándares de código resultan importantes en cualquier proyecto de desarrollo. Los mismos ayudan a asegurar que el código tenga una alta calidad, menos errores, pueda ser mantenido fácilmente y sea legible. Se utilizan los estándares de codificación definidos por Drupal en (Drupal.org, 2016). A continuación se explican y ejemplifican los estándares de codificación utilizados en la propuesta de solución.

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 (Gil, 2011). En el siguiente ejemplo se muestra un fragmento de código con las indentaciones realizadas:

```
188     <?php if (!empty($tabs)): ?>
189         <?php print render($tabs); ?>
190     <?php endif; ?>
```

Ilustración 6: Ejemplo de Identación

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; ?>`.]

```
290     <div class="imagen"></div>
291     <div class="titulo"><?php print $titulo; ?></div>
292     <div class="descripcion"><?php print $descripcion; ?></div>
```

Ilustración 7: Ejemplo de uso de punto y coma en código PHP

Uso de comillas: se pueden utilizar tanto las comillas simples como la ('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 las comillas dobles cuando el texto puede incluir alguna comilla simple.

```
310 ▼     <div class="grande">
311         <div class="imagen"></div>
312         <div class="titulo"><?php print $titulo; ?></div>
313         <div class="descripcion"><?php print $descripcion; ?></div>
314     </div>
```

Ilustración 8: Ejemplo de uso de comillas

Estructuras de control: para utilizar estas estructuras hay que tener en cuenta cuatro normas, las cuales definen el procedimiento a seguir. La primera norma es que debe haber un espacio entre el comando que define la estructura (*if*, *while* y *for*) y el paréntesis de apertura. La segunda norma es que la llave de apertura "{" se situarán en la misma línea que la definición de la estructura, separada por un espacio. La tercera es que se recomienda usar siempre las llaves "{}" aún en los casos en que no sea obligatorio su uso. La cuarta y última norma expresa que las estructuras **else** y **elseif** se escribirán en la línea siguiente al cierre de la sentencia anterior.

```

277     <?php if (!empty($arreglo_noticias)) { /* noticias */
278     ?>
279     <div class="mceItemTable contenedor" border="0" align="center">
280     <div class="header-block" style="width: 100%; text-align: center"><?php echo t('Noticias '); ?></div>
281
282
283     <?php
284     foreach ($arreglo_noticias as $noticia) {
285     $titulo = l($noticia->title, $base_url . "/node/" . $noticia->nid);
286     $descripcion = $noticia->field_desc_resu['und'][0]['value'];
287     $src = file_create_url($noticia->field_imagen['und'][0]['uri']);
288     ?>
289     <div class="grande">
290     <div class="imagen"></div>
291     <div class="titulo"><?php print $titulo; ?></div>
292     <div class="descripcion"><?php print $descripcion; ?></div>
293     </div>
294     <?php } ?>

```

Ilustración 9: Ejemplo de estructuras de control

3.3. Validación de la propuesta de solución

El aseguramiento de la calidad del *software* se ha convertido en una necesidad prioritaria y en una tarea vital en el desarrollo de cualquier sistema informático por la necesidad de garantizar que el producto cumpla con los requisitos especificados y que no presente errores. Por esta razón es necesario establecer un conjunto de pruebas para determinar el correcto funcionamiento del sistema, entre los principales tipos de pruebas realizadas a aplicaciones web se encuentran las pruebas de aceptación, las pruebas de usabilidad, las pruebas de seguridad y las pruebas de carga y estrés.

3.3.1. Pruebas funcionales

Pruebas de *software* que tienen por objetivo probar que los sistemas desarrollados cumplan con las funciones específicas para los cuales han sido creados. El enfoque de este tipo de prueba se basa en el análisis de los datos de entrada y en los de salida (López, 2012).

Tabla 5: Caso de prueba: Crear aviso

Escenario	Descripción	Título	Imagen	Descripción	Respuesta del sistema	Flujo central
EC 33.1 Crear aviso	El usuario con rol editor introduce los datos (título, imagen y descripción) correctos. El sistema permite que los usuarios con rol de administrador creen avisos.	Bien	Bien	Bien	El sistema verifica que no existan campos vacíos y los datos estén correctos, de ser así agrega el aviso.	El usuario con rol editor selecciona el enlace agregar contenido en la barra de menú, selecciona el contenido aviso, agrega todos los datos correctamente y selecciona el botón "Guardar".
		Jornada	Foto.jpg	prueba		
EC 33.2 Crear aviso incorrectamente	El usuario introduce el dato usuario incorrectamente.	Bien	Incorrecto	Vacío	El sistema verifica que los datos estén correctos, de no ser así, se muestra un mensaje de	El usuario selecciona el contenido Aviso, introduce los datos y selecciona el
		Tesis	Yo.mp4			

					error.	botón "Guardar".
--	--	--	--	--	--------	---------------------

Tabla 6: Descripción de las variables de Crear aviso

No	Nombre de campo	Clasificación	Valor Nulo	Descripción
1	Título	Campo de texto	no	Permite cadena de caracteres alfanuméricos (letra, números y caracteres especiales).
2	Descripción	Campo de texto	no	Permite cadena de caracteres alfanuméricos (letra, números y caracteres especiales).
	Imagen	Imagen	no	Campo para cargar imagen. Extensiones permitidas: jpg, jpeg, png.

Tabla 7: Caso de prueba: Crear vínculos bibliotecarios

Escenario	Descripción	Título	Descripción	URL	Respuesta del sistema	Flujo central
EC 33.1	El usuario con	Bien	Bien	Bien	El sistema	El usuario con

<p>Crear vínculos bibliotecarios</p>	<p>rol editor introduce los datos (título, imagen y descripción) correctos.</p> <p>El sistema permite que los usuarios con rol de administrador creen los vínculos bibliotecarios.</p>	<p>Centro Infomed</p>	<p>Portal web de Ciencias Medicas</p>	<p>www.info med.sld.cu</p>	<p>verifica que no existan campos vacíos y los datos estén correctos, de ser así agrega el vínculo bibliotecario.</p>	<p>rol editor selecciona el enlace "agregar contenido" en la barra de menú, selecciona el contenido "vínculos bibliotecarios" , agrega todos los datos correctamente y selecciona el botón "Guardar".</p>
<p>EC 33.2 Crear vínculos bibliotecarios incorrectamente</p>	<p>El usuario introduce el dato usuario incorrectamente.</p>	<p>Bien</p> <hr/> <p>UCI</p>	<p>Vacío</p> <hr/>	<p>Incorrecto</p> <hr/> <p>uci</p>	<p>El sistema verifica que los datos estén correctos de no ser así, se muestra un mensaje de error.</p>	<p>El usuario selecciona el contenido "vínculos bibliotecarios" , introduce los datos y selecciona el botón "Guardar".</p>

Tabla 8: Descripción de las variables Crear vínculos bibliotecarios

No	Nombre de campo	Clasificación	Valor Nulo	Descripción
1	Título	Campo de texto	no	Permite cadena de caracteres alfanuméricos (letra, números y caracteres especiales).
2	Descripción	Campo de texto	no	Permite cadena de caracteres alfanuméricos (letra, números y caracteres especiales).
	URL	Campo de texto	no	Permite añadir un enlace: debe ponerse una URL al sitio para el cual se requiere hacer el enlace.

Resultados de las pruebas a las funcionalidades del software

Las pruebas funcionales aplicadas a la propuesta de solución fueron basadas en el diseño de casos de pruebas, donde se recogen los escenarios correspondientes a cada requisito funcional. En correspondencia con los artefactos generados por la metodología AUP. Para la realización de las pruebas se empleó la plantilla de casos de pruebas, en la que se describe el comportamiento de la aplicación ante determinado flujo de datos especificado por el probador.

En la primera iteración se detectan un total de 36 no conformidades, de ellas 16 fueron de validación, 7 de ortografía, 3 funcionales, 5 de redacción y 5 de idioma. Las 16 de validación fueron tipos de archivos y fueron resueltas definiendo sus extensiones, las 7 de ortografía y 5 de redacción se corrigieron editando los tipos de contenidos donde se encontraban ubicados, las 5 de idiomas fueron resueltas a través de la traducción de las interfaces y de las 3 funcionales solo fue resuelta la autenticación con el sistema LDAP con la configuración del módulo LDAP, quedando pendiente el envío de correo del boletín informativo y el contacto bibliotecario.

En la segunda iteración se encuentran un total de 12 no conformidades, las 2 pendientes de la iteración anterior que fueron resueltas con los módulos SMTP y *Mail System*, generando 10 conformidades nuevas de idioma, las cuales fueron resueltas con la traducción de las interfaces. Como resultado final todas las no conformidades encontradas fueron resueltas.

En la tercera iteración no se encontraron no conformidades lo que demuestra que el *software* funciona correctamente.

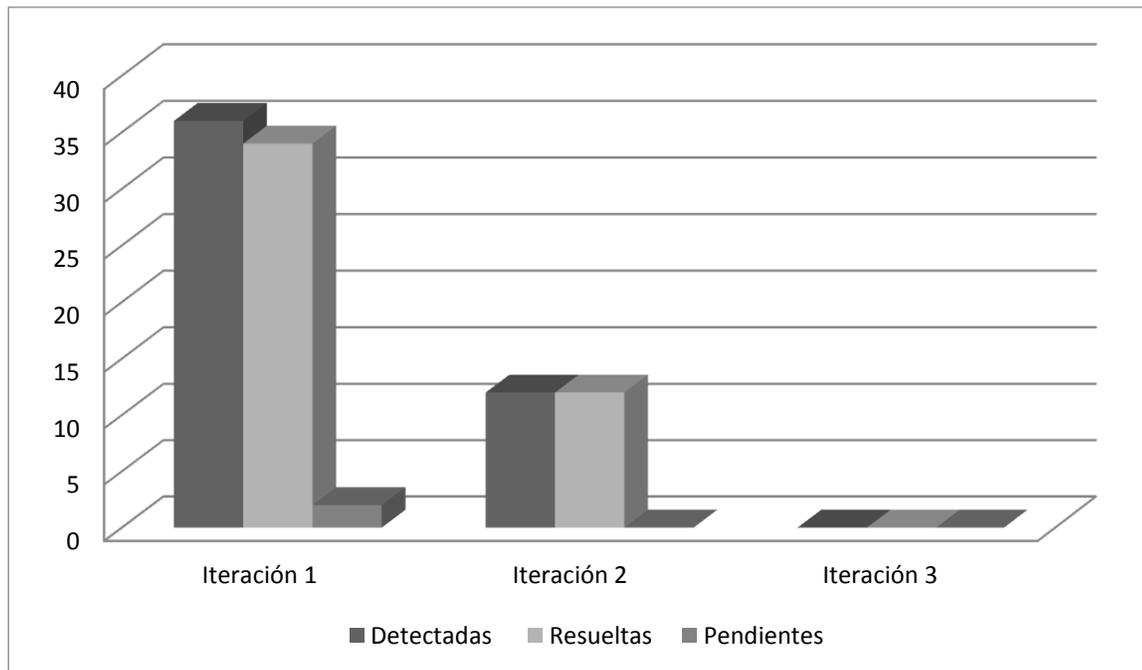


Ilustración 10: Comportamiento de las no conformidades por iteraciones

3.3.2. Pruebas de seguridad

La seguridad es uno de los factores importantes para la calidad del *software*. El *software* es totalmente seguro cuando se comporta de una manera correcta ante la presencia de ataques maliciosos (Kumar, 2014). Para garantizar la seguridad del *software* es aplicado el proceso de pruebas de seguridad.

Resultado de las pruebas de seguridad

Para evaluar la seguridad de la aplicación se hizo uso de la herramienta Acunetix WVS v8 la cual devolvió los siguientes resultados:

Tabla 9: Resultado de la aplicación de la herramienta Acunetix WVS 8 para pruebas de seguridad

Categorías de vulnerabilidades	Cantidad
La función allow_url_fopen de PHP está activada	1
La función open_basedir de PHP no está configurada	1

El allow_url_fopen directiva de configuración de PHP está habilitado. Cuando está activada, esta directiva permite la recuperación de datos desde ubicaciones remotas (sitio web o servidor FTP). Un gran número de vulnerabilidades de inyección de códigos reportados en PHP son basadas en aplicaciones web causadas por la combinación de habilitar el allow_url_fopen y un mal filtrado de entrada.

La directiva de configuración open_basedir limitará los archivos que pueden ser abiertos por PHP al árbol de directorios especificado. Cuando una secuencia de comandos intenta abrir un archivo como por ejemplo, fopen () o gzopen (), se comprueba la ubicación del archivo. Cuando el archivo está fuera del árbol de directorios especificado, PHP se negará a abrirlo. La función open_basedir es una buena protección contra la inclusión de vulnerabilidades por archivo remoto. Para un atacante remoto no es posible romper las restricciones de open_basedir, si el solamente es capaz de introducir el nombre de un archivo para ser incluido. Por lo tanto el número de archivos que va a ser capaz de incluir con tal de insertar vulnerabilidades a archivo locales es limitado.

Se corrigieron las vulnerabilidades relacionadas con las funciones de PHP mediante la configuración del archivo php.ini, deshabilitando la función allow_url_fopen y a la función open_basedir la ubicación del directorio donde se encuentra alojado el portal web.

Las vulnerabilidades encontradas fueron problemas del servidor web donde se hallaba alojado el sitio web y fueron resueltas a través de la configuración del propio servidor.

3.3.3. Pruebas de rendimientos

Según la IEEE (*Institute of Electrical and Electronics Engineers*): “Las pruebas de rendimientos es el grado en que un sistema o componente realiza sus funciones designadas dentro de las limitaciones dadas, tales como la velocidad, precisión o el uso de la memoria”(TESTING EN ESPAÑOL, 2011).

Pruebas de carga: mediante la ejecución de las pruebas de carga es posible identificar la capacidad de recuperación de un sistema cuando es sometido a cargas variables tanto de usuarios como de procesos. Al realizar las pruebas de carga, se puede determinar el tiempo de respuesta de todas las transacciones críticas del sistema y encontrar cuellos de botella de la aplicación (V&v Quality S.A., 2016).

Pruebas de estrés: estas pruebas son utilizadas normalmente para someter a la aplicación al límite de su funcionamiento mediante la ejecución de un número de usuarios muy superior al esperado. Esta prueba tiene como finalidad el determinar la robustez de una aplicación cuando la carga es extrema y ayuda a administradores a determinar si la aplicación se comportará debidamente ante diferentes situaciones (Zapata y Velázquez, 2011).

Resultados de las pruebas de rendimiento

Para la realización de las pruebas de carga y estrés se empleó la herramienta Apache JMeter. El ambiente de prueba estuvo conformado por:

- Sistema Operativo: Linux Mint 17.2
- Microprocesador: AMD Athlon 64 @ 1.70GHz x2.
- Memoria RAM: 2 GB
- Disco Duro: 250 GB

Los resultados de las pruebas de carga y estrés se consideran satisfactorios, debido a que los tiempos de respuesta del servidor ante la interacción de 100 usuarios concurrentes se encuentra en el rango de tiempo de 1 a 3 segundos. La propuesta de solución generó un total de 46.7 kb/segundos de transferencia de datos para 100 usuarios, con un rendimiento de 6.1 segundos. Con ello queda demostrado que la propuesta de solución es estable, ya que se mantuvo prestando servicios todo el tiempo, sin incurrir en fallos como muestra la siguiente imagen:

Etiqueta	# Muestras	Media	Mediana	Linea de 90%	Mín	Máx	% Error	Rendimiento	Kb/sec
78 /ober/	100	21541	24344	28717	1873	31056	0,00%	3,1/sec	28,8
82 /ober/	100	14070	14998	15389	2655	15741	0,00%	2,2/sec	16,8
86 /ober/	100	15137	15297	15553	11352	15687	0,00%	1,7/sec	12,3
91 /ober/	100	15332	15364	15753	14526	16020	0,00%	1,6/sec	11,7
95 /ober/	100	14270	14365	15436	12821	15972	0,00%	1,6/sec	11,9
99 /ober/	100	11753	11914	15023	7110	15735	0,00%	1,9/sec	13,8
Total	600	15351	15109	22007	1873	31056	0,00%	6,1/sec	46,7

Ilustración 11: Resultado de las pruebas de carga y estrés

3.3.4. Pruebas de usabilidad

La meta de esta prueba de usabilidad es identificar y rectificar las deficiencias de usabilidad y errores existentes. Se intenta asegurar que la aplicación sea fácil de aprender y de usar, su uso sea satisfactorio, provea utilidad y funcionalidad y cumpla las tareas para las cuales fue desarrollada la aplicación (Greiner y Petris 2012)

Para la realización de las pruebas de usabilidad se utilizó una lista de chequeo, que se puede consultar en los anexos, que divide las preguntas a realizar en las siguientes categorías:

- Visibilidad del sistema.
- Lenguaje común entre sistema y usuario.
- Libertad y control por parte del usuario.
- Consistencia y estándares.
- Estética y diseño minimalista.
- Prevención de errores.
- Ayuda a los usuarios a reconocer, diagnosticar y recuperarse de los errores.

- Ayuda y documentación.

Se realizó una iteración de las pruebas que arrojó un total de 7 NC de las cuales 2 no procedieron ya que no se encontraban entre los requerimientos del cliente y las otras se dividieron de la siguiente manera:

Tabla 10: Resultados de las pruebas de usabilidad

Categoría	Cantidad	No Conformidad
Visibilidad del sistema	3	Cada pantalla no empieza con un título que describe su contenido
		Cuándo se selecciona un icono ¿se diferencia de los no seleccionados?
		¿No utiliza más de siete opciones principales en el menú de navegación?
Consistencia y estándares	1	¿Añade una descripción en las imágenes?
Prevención de errores	1	¿Se dan indicaciones para completar campos problemáticos?

Todas las no conformidades detectadas fueron resueltas y verificadas en una regresión realizada.

Conclusiones parciales

La deficiente realización de pruebas de *software* a los sistemas de cómputo, constituyen hoy en día una de las causas de mayor impacto que provoca la baja calidad de las aplicaciones informáticas. Por esta razón, la ejecución de las pruebas al portal web del CICT permitió detectar las deficiencias presentes, subsanarlas en el menor tiempo posible y ofrecer una aplicación con mayor calidad, seguridad y usabilidad.

Conclusiones generales

Al término de la fundamentación teórica en la que se sustentó la presente investigación y del desarrollo y validación del portal web del Centro de Información Científico Técnica, se arriba a las siguientes conclusiones:

- La investigación demostró la necesidad de desarrollar un portal web para el Centro de Información Científico-Técnica de la Universidad de las Ciencias Informáticas, que permita mejorar la gestión de información de los recursos informativos.
- La definición de los requisitos funcionales y no funcionales permitió obtener el diseño y la realización de la propuesta de solución a la problemática planteada.
- La implementación de un tema para Drupal y la utilización de diferentes módulos necesarios para el funcionamiento de la propuesta de solución contribuyó a obtener un portal web que permita mejorar la gestión de los recursos informativos del Centro de Información Científico-Técnica.
- Aplicadas las técnicas de validación a la propuesta de solución, se comprobó el buen funcionamiento del sistema de acuerdo a los requisitos planteados.

Recomendaciones

Una vez concluida la investigación y el desarrollo de la propuesta de solución, el autor del presente trabajo recomienda implementar un módulo que permita realizar la búsqueda desde el portal web para el Centro de Información Científico Técnica en los materiales registrados en el Catálogo y el Repositorio Institucional de la Universidad de las Ciencias Informática.

Referencias bibliográficas

- ACUNETIX. Web application security with Acunetix. [en línea]. 2016, [Consulta: 10 mayo 2016]. Disponible en: <http://www.acunetix.com/>.
- ADAMO, O.I. Inf. de teoría de la inf. programación. Scribd [en línea], 2011. [Consulta: 18 junio 2016]. Disponible en: <https://es.scribd.com/doc/47268717/INF-DE-TEORIA-DE-LA-INF-PROGRAMACION>.
- ALVAREZ, S. Sistemas gestores de bases de datos. [en línea]. 2007, [Consulta: 6 diciembre 2015]. Disponible en: <http://www.desarrolloweb.com/articulos/sistemas-gestores-bases-datos.html>.
- APACHE JMETER. Apache JMeter - Apache JMeter™. [en línea]. 2015, [Consulta: 10 mayo 2016]. Disponible en: <http://jmeter.apache.org/>.
- BIBLIOTECA UNIVERSIDAD DE ALCALÁ. Fuentes de información. [en línea]. 2015, [Consulta: 24 mayo 2016]. Disponible en: <http://www3.uah.es/bibliotecaformacion/BPOL/FUENTESDEINFORMACION/>.
- BRAVO, S. y GARCÍA, T.D. Portal de trámites de la República de Cuba. [en línea]. 2014, [Consulta: 25 mayo 2016]. Disponible en: http://repositorio_institucional.uci.cu//jspui/handle/ident/8125.
- BURK, C.F. y HORTON, F.W. *Infomap: A Complete Guide to Discovering Corporate Information Sources*, 1991. 24p.
- CHOO, C.W. *Information Management for the Intelligent Organization: The Art of Scanning the Environment*. S.l.: Information Today, Inc, 2002.
- CODINA, L. Evaluación de calidad en sitios web: metodología de proyectos de análisis sectoriales y de realización de auditorías (Evaluation of quality in web sites: projects methodology for sectorial analyses and audits), 2007.
- CODING STANDARDS | Drupal.org. [en línea], 2016. [Consulta: 14 marzo 2016]. Disponible en: <https://www.drupal.org/coding-standards>.
- COWBURN, P. PHP: ¿Qué puede hacer PHP? - Manual. [en línea], 2014. [Consulta: 6 diciembre 2015]. Disponible en: <http://php.net/manual/es/intro-whatcando.php>.
- DOMÍNGUEZ, A. *MANUAL BÁSICO DE WORDPRESS* [en línea], 2007. [Consulta: 6 diciembre 2015]. Disponible en: <https://carmenvera.wikispaces.com/file/view/manual+WordPress.pdf>.
- EGUILUZ, J. Capítulo 1. Introducción (Introducción a CSS). [en línea], 2007. Disponible en: https://librosweb.es/libro/css/capitulo_1.html.

- GENBETA DEV. NetBeans. [en línea], 2014. [Consulta: 6 diciembre 2015]. Disponible en: <http://www.genbetadev.com/herramientas/netbeans-1>.
- GÓMEZ, J.C.G. Portales en Internet: concepto, tipología básica y desarrollo. *El profesional de la información*, 2011, 10: p. 4-13.
- GREINER, C.L. y PETRIS, R.H. *Calidad de software e ingeniería de usabilidad* [en línea], 2012. [Consulta: 26 mayo 2016]. Disponible en: http://sedici.unlp.edu.ar/bitstream/handle/10915/19202/Documento_completo.pdf?sequence=1.
- KEMPPAINEN, T.-P. *Data Archive Project*. Helsinki: Helsinki Metropolia University of Applied Sciences, 2015.
- KUMAR, R. *Software Security Testing A Pertinent Framework*. *Journal of Global Research in Computer Science*, 2014, 5 (3), p. 23-27.
- LÓPEZ, A.G. Pruebas funcionales. [en línea], 2012. [Consulta: 26 mayo 2016]. Disponible en: <http://es.slideshare.net/AlfonsoLopezceballos/pruebas-funcionales>.
- MARTÍNEZ, B.E.V. *Panorama general de las «Herramientas CASE»* [en línea], 2007. [Consulta: 6 diciembre 2015]. Disponible en: <http://dgsa.uaeh.edu.mx:8080/bibliotecadigital/bitstream/handle/231104/1669/Panorama%20genera%20de%20las%20herramientas%20CASE.pdf?sequence=1>.
- MORALES, P.A.A. Servidores Web. [en línea], 2015. [Consulta: 6 diciembre 2015]. Disponible en: <http://www.monografias.com/trabajos75/servidores-web/servidores-web.shtml>.
- MOSCOSO, P. Reflexiones en torno al concepto «recurso de información». *Revista General de Información y Documentación*, 1998, 8 (1), p. 327.
- PEDRE. Aspectos Teóricos - Entorno de Desarrollo Integrado (IDE). [en línea], 2012. [Consulta: 18 junio 2016]. Disponible en: <http://cu.globedia.com/aspectos-teoricos-entorno-desarrollo-integrado-ide>.
- PLAYÁN, J.L. *Entorno de audio usando la nueva API de HTML5*. [en línea], 2014. [Consulta: 6 diciembre 2015]. Disponible en: https://riunet.upv.es/bitstream/handle/10251/46580/latorre_playan_javier.pdf?sequence=1.
- PONJUÁN, G. *Gestión de información. Dimesiones e implementación para el éxito organizacional*. Trea, 2007.
- POSTGRESQL. PostgreSQL: About. [en línea], 2010. [Consulta: 6 diciembre 2015]. Disponible en: <http://www.postgresql.org/about/>.
- PRYTHERCH, R.J. *Harrod's Librarians' Glossary and Reference Book: A Directory of Over 9,600 Terms, Organizations, Projects, and Acronyms in the Areas of Information Management, Library Science, Publishing, and Archive Management*. Gower, 2000.

- RANCEL, M.R. ¿Qué es y para qué sirve *Joomla*? CMS potente e interesante para crear páginas web de todo tipo. [en línea], 2010. [Consulta: 6 diciembre 2015]. Disponible en: http://aprenderaprogramar.com/index.php?option=com_content&view=article&id=119:ique-es-y-para-que-sirve-Joomla-cms-potente-e-interesante-para-crear-paginas-web-de-todo-tipo-cu00403a&catid=38:curso-qcreacion-web-con-Joomla-desde-ceroq&Itemid=152.
- SÁNCHEZ, J. *JavaScript manual de referencia* [en línea], 2003. [Consulta: 6 diciembre 2015]. Disponible en: <http://www.jorgesanchez.net/web/JavaScript.pdf>.
- SÁNCHEZ, T.R. *Metodología de desarrollo para la Actividad productiva de la UCI*. 2015.
- SIERRA, D. *Visual Paradigm For Uml*. [en línea], 2007. [Consulta: 6 diciembre 2015]. Disponible en: <http://es.slideshare.net/vanquishdarkenigma/visual-paradigm-for-uml>.
- URIBE, C.A.C., MONTOYA, C.E.G. y RODRÍGUEZ, L.E.S., 2015. *Balanceo de Carga de Aplicaciones Web con Apache Web Server* [en línea], 2015. [Consulta: 6 diciembre 2015]. Disponible en: <http://repositorio.uniquindio.edu.co/handle/123456789/486>.
- VERMA, J., PANDEY, H. y BANSAL, S. *Development of Micro Aerial Vehicle using Piezoelectric Fiber Composites* [en línea], 2014. [Consulta: 26 mayo 2016]. Disponible en: <http://www.ijser.org/researchpaper\Development-of-Micro-Aerial-Vehicle-using-Piezoelectric-Fiber-Composites.pdf>.

Bibliografía

- ACUNETIX. Web application security with Acunetix. [en línea]. 2016, [Consulta: 10 mayo 2016]. Disponible en: <http://www.acunetix.com/>.
- ADAMO, O.I. Inf. de teoría de la inf. programación. *Scribd* [en línea], 2011. [Consulta: 18 junio 2016]. Disponible en: <https://es.scribd.com/doc/47268717/INF-DE-TEORIA-DE-LA-INF-PROGRAMACION>.
- ALVAREZ, M.A. Desarrollo Web, Tu mejor ayuda para aprender hacer webs. [en línea], 2009. [Consulta: 21 octubre 2015]. Disponible en: <http://www.desarrolloweb.com/articulos/392.php>.
- ALVAREZ, S. Sistemas gestores de bases de datos. [en línea]. 2007, [Consulta: 6 diciembre 2015]. Disponible en: <http://www.desarrolloweb.com/articulos/sistemas-gestores-bases-datos.html>.
- APACHE JMETER. Apache JMeter - Apache JMeterTM. [en línea]. 2015, [Consulta: 10 mayo 2016]. Disponible en: <http://jmeter.apache.org/>.
- BIBLIOTECA UNIVERSIDAD DE ALCALÁ. Fuentes de información. [en línea]. 2015, [Consulta: 24 mayo 2016]. Disponible en: <http://www3.uah.es/bibliotecaformacion/BPOL/FUENTESDEINFORMACION/>.
- BRAVO, S. y GARCÍA, T.D. Portal de trámites de la República de Cuba. [en línea]. 2014, [Consulta: 25 mayo 2016]. Disponible en: http://repositorio_institucional.uci.cu//jspui/handle/ident/8125.
- BURK, C.F. y HORTON, F.W. Infomap: A Complete Guide to Discovering Corporate Information Sources, 1991. 24p.
- CHOO, C.W. Information Management for the Intelligent Organization: The Art of Scanning the Environment. S.I.: Information Today, Inc, 2002.
- CODINA, L. Evaluación de calidad en sitios web: metodología de proyectos de análisis sectoriales y de realización de auditorías (Evaluation of quality in web sites: projects methodology for sectorial analyses and audits), 2007.
- CODING STANDARDS | Drupal.org. [en línea], 2016. [Consulta: 14 marzo 2016]. Disponible en: <https://www.drupal.org/coding-standards>.
- COWBURN, P. PHP: ¿Qué puede hacer PHP? - Manual. [en línea], 2014. [Consulta: 6 diciembre 2015]. Disponible en: <http://php.net/manual/es/intro-whatcando.php>.
- DOMÍNGUEZ, A. MANUAL BÁSICO DE WORDPRESS [en línea], 2007. [Consulta: 6 diciembre 2015]. Disponible en: <https://carmenvera.wikispaces.com/file/view/manual+WordPress.pdf>.
- EGUILUZ, J. Capítulo 1. Introducción (Introducción a CSS). [en línea], 2007. Disponible en: https://librosweb.es/libro/css/capitulo_1.html.

- GENBETA DEV. NetBeans. [en línea], 2014. [Consulta: 6 diciembre 2015]. Disponible en: <http://www.genbetadev.com/herramientas/netbeans-1>.
- GÓMEZ, J.C.G. Portales en Internet: concepto, tipología básica y desarrollo. El profesional de la información, 2011, 10: p. 4-13.
- GREINER, C.L. y PETRIS, R.H. Calidad de software e ingeniería de usabilidad [en línea], 2012. [Consulta: 26 mayo 2016]. Disponible en: http://sedici.unlp.edu.ar/bitstream/handle/10915/19202/Documento_completo.pdf?sequence=1.
- KEMPPAINEN, T.-P. Data Archive Project. Helsinki: Helsinki Metropolia University of Applied Sciences, 2015.
- KUMAR, R. Software Security Testing A Pertinent Framework. Journal of Global Research in Computer Science, 2014, 5 (3), p. 23-27.
- LIMONES, W.D. Desarrollo de portal web para administración remota de servicios, logs y servidores sobre plataforma Windows [en línea], 2015. [Consulta: 26 mayo 2016]. Disponible en: <http://repositorio.ug.edu.ec/bitstream/redug/6998/1/LIMONES%20ESCOBAR%20WELLINGTON%20ODANIXER.pdf>.
- LÓPEZ, A.G. Pruebas funcionales. [en línea], 2012. [Consulta: 26 mayo 2016]. Disponible en: <http://es.slideshare.net/AlfonsoLopezCeballos/pruebas-funcionales>.
- MARTÍNEZ, B.E.V. Panorama general de las «Herramientas CASE» [en línea], 2007. [Consulta: 6 diciembre 2015]. Disponible en: <http://dgsa.uaeh.edu.mx:8080/bibliotecadigital/bitstream/handle/231104/1669/Panorama%20genera%20de%20las%20herramientas%20CASE.pdf?sequence=1>.
- MORALES, P.A.A. Servidores Web. [en línea], 2015. [Consulta: 6 diciembre 2015]. Disponible en: <http://www.monografias.com/trabajos75/servidores-web/servidores-web.shtml>.
- MOREIRA, G.I. Sistema web para seguimiento de una comunidad de ex alumnos en un centro educativo. [en línea], 2015. [Consulta: 27 mayo 2016]. Disponible en: <http://repositorio.ug.edu.ec/handle/redug/10117?mode=full>.
- MOSCOSO, P. Reflexiones en torno al concepto «recurso de información». Revista General de Información y Documentación, 1998, 8 (1), p. 327.
- PEDRE. Aspectos Teóricos - Entorno de Desarrollo Integrado (IDE). [en línea], 2012. [Consulta: 18 junio 2016]. Disponible en: <http://cu.globedia.com/aspectos-teoricos-entorno-desarrollo-integrado-ide>.
- PERFORMANCE TESTING. Testing en Español [en línea], 2009. [Consulta: 5 abril 2016]. Disponible en: <https://josepablosarco.WordPress.com/performance-testing/>.

- PEROTTO, H.L. y RIVERO, K. Distributing biodiversity data through the web: The Geospatial Center for Biodiversity in Bolivia. Department of Environmental Science and Technology. Cranfield University, 2012.
- PLAYÁN, J.L. Entorno de audio usando la nueva API de HTML5. [en línea], 2014. [Consulta: 6 diciembre 2015]. Disponible en: https://riunet.upv.es/bitstream/handle/10251/46580/latorre_playan_javier.pdf?sequence=1.
- PONJUÁN, G. Gestión de información. Dimesiones e implementación para el éxito organizacional. Trea, 2007.
- POSTGRESQL. PostgreSQL: About. [en línea], 2010. [Consulta: 6 diciembre 2015]. Disponible en: <http://www.postgresql.org/about/>.
- PRYTHERCH, R.J. Harrod's Librarians' Glossary and Reference Book: A Directory of Over 9,600 Terms, Organizations, Projects, and Acronyms in the Areas of Information Management, Library Science, Publishing, and Archive Management. Gower, 2000.
- RANCEL, M.R. ¿Qué es y para qué sirve Joomla? CMS potente e interesante para crear páginas web de todo tipo. [en línea], 2010. [Consulta: 6 diciembre 2015]. Disponible en: http://aprenderaprogramar.com/index.php?option=com_content&view=article&id=119:ique-es-y-para-que-sirve-Joomla-cms-potente-e-interesante-para-crear-paginas-web-de-todo-tipo-cu00403a&catid=38:curso-qcreacion-web-con-Joomla-desde-ceroq&Itemid=152.
- RODRÍGUEZ, R., CODINA, L. y PEDRAZA, R. Indicadores para la evaluación de la calidad en cibermedios: análisis de la interacción y de la adopción de la Web 2.0. Revista española de Documentación Científica, 2012, 35 (1): p. 61-93.
- SÁNCHEZ, J. JavaScript manual de referencia [en línea], 2003. [Consulta: 6 diciembre 2015]. Disponible en: <http://www.jorgesanchez.net/web/JavaScript.pdf>.
- SÁNCHEZ, T.R. Metodología de desarrollo para la Actividad productiva de la UCI. 2015.
- SIERRA, D. Visual Paradigm For Uml. [en línea], 2007. [Consulta: 6 diciembre 2015]. Disponible en: <http://es.slideshare.net/vanquishdarkenigma/visual-paradigm-for-uml>.
- URIBE, C.A.C., MONTOYA, C.E.G. y RODRÍGUEZ, L.E.S. Balanceo de Carga de Aplicaciones Web con Apache Web Server [en línea], 2015. [Consulta: 6 diciembre 2015]. Disponible en: <http://repositorio.uniquindio.edu.co/handle/123456789/486>.
- VERMA, J., PANDEY, H. y BANSAL, S. Development of Micro Aerial Vehicle using Piezoelectric Fiber Composites [en línea], 2014. [Consulta: 26 mayo 2016]. Disponible en: <http://www.ijser.org/researchpaper\Development-of-Micro-Aerial-Vehicle-using-Piezoelectric-Fiber-Composites.pdf>.

Glosario de términos

CASE: Ingeniería de Software Asistida por Ordenador (siglas del inglés *Computer Aided Software Engineering*),

CICT: Centro de Información Científico-Técnica

CMS: Sistema Gestor de Contenido *Content Management System*

CPU: Unidad de Procesamiento Central (siglas del inglés *Central Processing Unit*)

CRAI: Centro de Recursos de Aprendizaje y la Investigación

CSS: Hojas de estilo en cascada (siglas del inglés *Cascading Style Sheets*)

GNU: acrónimo Gnu no es Unix (siglas del inglés Gnu Not Unix)

GPL: Licencia Pública General (siglas del inglés *General Public License*)

HTTP: Protocolo de Transferencia de Hipertexto (siglas del inglés *Hypertext Transfer Protocol*)

LDAP: Protocolo Ligero de Acceso a Directorios (siglas del inglés *Lightweight Directory Access Protocol*)

ODBC: *Open DataBase Connectivity*

PHP: Procesador de Hipertexto (siglas del inglés Hypertext Preprocessor)

SMTP: Protocolo de transferencia Simple de Correo (siglas del inglés *Simple Mail Transfer Protocol*)

Tcl: *Tool Command Language*

TCP: Protocolo de Control de Transmisión (TCP siglas del inglés *Transfer Control Protocol*)

TIC: Tecnologías de la Información y la Comunicación

UCI: Universidad de las Ciencias Informáticas

Webmaster: Contracción de las palabras inglesas *web* y *master*

XHTML: *eXtensible HyperText Markup Language*

XML: *eXtensible Markup Language*