



Facultad I

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

Título:

*Módulo para la creación de contenido web vía
correo electrónico en el portal de Cuba Sí
desarrollado en Drupal.*

Autor:

Bárbara Beatriz Somoza Leal

Tutores:

Ing. Marilys Valiente González

Ing. Olber Ríos Cutiño

La Habana, junio 2017



“Seamos realistas, hagamos lo imposible.”

Ernesto “Che” Guevara de la Serna.

Declaración de autoría

Declaro por este medio que yo Bárbara Beatriz Somoza Leal, con carné de identidad 94120604996 soy el autor principal del trabajo titulado “Módulo para la creación de contenido web vía correo electrónico en el portal de CubaSí desarrollado en Drupal” y autorizo a la Universidad de las Ciencias Informáticas a hacer uso de la misma en su beneficio, así como los derechos patrimoniales con carácter exclusivo.

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

Firma del Autor
Bárbara Beatriz Somoza Leal

Firma del Tutor
Ing. Marilys Valiente González

Firma del Tutor
Ing. Olber Ríos Cutiño

Dedicatoria

A mi mamá que ha sido madre, padre y amiga, hoy me siento orgullosa de haber logrado juntas este objetivo y de tenerte como guía, gracias mami por todo el amor y dedicación.

A mi abuela, aunque no estés sé que estarías muy orgullosa de mi, hoy tu nieta se hace ingeniera, hoy sé que estas muy feliz por mí.

A Dani (mi titi), gracias a ti comprendí el verdadero significado del amor, estoy muy agradecida por todos los bellos momentos que hemos compartido, por todas las alegrías, por haberme elegido para acompañarte y amarte.

Agradecimientos

A mi familia por estar siempre presente, a toda la familia de Dani, hoy los considero mi familia también, gracias por el amor incondicional que me han dado.

A mis profesores por el apoyo brindado en el transcurso de mi carrera, a mis tutores Marylis y Olber por creer en mí, al tribunal gracias por dedicarme su tiempo, a mis compañeros de aula por estos 5 años de apoyo y esfuerzo, en especial a Laura por soportarme.

A las chicas del 110-104 Wilbia, Betsy, Raiza y Lisbet gracias por su amistad.

A todos los amigos que he conocido en estos 5 años en esta universidad, en especial a la gente del piquete Adrian, David, Aldo, Angie, etc, los quiero mucho, gracias por ser parte de la persona que hoy soy.

Resumen

El Centro de Ideoinformática (CIDI), perteneciente a la Facultad 1 de la Universidad de las Ciencias Informáticas (UCI) entre las líneas de investigación y desarrollo del centro se encuentra la producción de portales web. Uno de los sitios web desarrollados por el centro es CubaSí en el cual se identificó la necesidad de integrar una funcionalidad que permita a los usuarios autenticados en el portal, publicar contenido web a través de una cuenta de correo electrónico única, por lo que se decide desarrollar un módulo para la publicación de contenidos web vía correo electrónico.

Se define AUP-UCI como metodología de desarrollo de software para guiar este proceso. El sistema fue diseñado e implementado a partir del estudio de algunas tecnologías y herramientas utilizadas en la actualidad para la construcción de módulos para el CMS Drupal en su versión 7. Además, fueron desarrolladas un conjunto de pruebas que arrojaron como resultado que la propuesta de solución es funcional, segura y se integra de forma adecuada al sistema gestor de contenido Drupal 7 y al portal web CubaSí.

El módulo para la publicación de contenidos web vía correo electrónico en el portal de CubaSí contribuye a la mejora del proceso de publicación de contenidos web, pues posibilita la publicación de imágenes y documentos a través de los correos, entre otros.

Palabras clave: correo electrónico, CubaSí, Drupal, módulo, publicación de contenido web.

Índice

Introducción	1
Capítulo I: Fundamentación teórica.....	8
1.1. Introducción	8
1.2. Conceptos asociados al módulo para la publicación de contenidos web vía correo electrónico.....	8
1.3. Sistemas homólogos.....	9
1.4. Metodología para el desarrollo de software.....	12
1.5. Sistema de gestión de contenidos	14
1.6. Herramientas para el desarrollo	15
1.6.1. Lenguaje de modelado	15
1.7. Conclusiones parciales	22
Capítulo II. Análisis y diseño del módulo para la publicación de contenidos web vía correo electrónico.....	23
2.1. Propuesta de solución	23
2.2. Modelo de dominio	24
2.3. Requisitos Funcionales	25
2.4. Requisitos no funcionales	30
2.5. Arquitectura de software	32
2.6. Patrones de diseño	32
2.7. Modelo de despliegue.....	34
2.8. Conclusiones parciales	37
Capítulo III. Implementación y validación del Módulo para la publicación de contenidos web vía correo electrónico	38
3.1. Introducción	38
3.2. Diagrama de componentes	38
3.3. Estándares de codificación	39
3.4. Validación de la propuesta de solución	44
3.5. Validación de la hipótesis.....	53
3.6. Conclusiones parciales	58
Conclusiones	59

Recomendaciones	60
Bibliografía.....	61
Anexos.....	66

Índice de figuras

Figura 1. Modelo de dominio.....	25
Figura 2. Características de la arquitectura basada en componentes	32
Figura 3. Modelo de despliegue	35
Figura 4. Diagrama de componentes	38
Figura 5. Ejemplo de indentación y espacios en blanco	39
Figura 6. Ejemplo de operadores	40
Figura 7. Ejemplo de estructuras de control	40
Figura 8. Ejemplo de la estructura if y else.....	40
Figura 9. Ejemplo de la estructura switch.....	41
Figura 10. Ejemplo de longitud de línea menor a 80 caracteres.....	41
Figura 11. Ejemplo de longitud de línea mayor a 80 caracteres	41
Figura 12. Ejemplo de llamadas a funciones.....	42
Figura 13. Ejemplo de array.....	42
Figura 14. Ejemplo de uso de comillas simples.....	42
Figura 15. Ejemplo de apertura y cierre de un archivo .inc.....	43
Figura 16. Ejemplo del terminador (;)	43
Figura 17. Ejemplo de nombre de funciones	43
Figura 18. Ejemplo de nombre de variables	43
Figura 19. Ejemplo de variable persistente	44

Figura 20. Ejemplo de constantes.....	44
Figura 21. Comportamiento de las no conformidades por iteraciones.....	48
Figura 22. Configuración para proteger archivos sensibles.....	51
Figura 23. Valores obtenidos para una muestra de 25 usuarios.....	52
Figura 24. Comportamiento de la valoración de los expertos según los parámetros evaluativos.....	57

Índice de tablas

Tabla 1. Operacionalización de las variables.....	3
Tabla 2. Comparación de los sistemas homólogos.....	12
Tabla 3. Requisitos funcionales.....	26
Tabla 4. H.U. Crear Cuenta.....	27
Tabla 5. H.U. Migrar contenido.....	29
Tabla 6. Caso de prueba Crear Cuenta.....	46
Tabla 7. Resultado del escaneo de vulnerabilidades para pruebas de seguridad.....	50
Tabla 8. Expertos seleccionados en la validación de la investigación.....	54
Tabla 9. Sentencias aplicadas a los expertos.....	55
Tabla 10. Frecuencias asociadas a los parámetros evaluativos.....	56
Tabla 11. H.U. Verificar datos de la cuenta.....	66
Tabla 12. H.U. Revisar los correos electrónicos.....	67
Tabla 13. Caso de prueba 2 Editar Cuenta.....	68
Tabla 14. Caso de prueba 5 Clonar Cuenta.....	70

Introducción

La sociedad actual cuenta con grandes avances en las Tecnologías de la Información y las Comunicaciones (TIC), destinados a lograr una mejora en el proceso de comunicación (Rodríguez Salas, et al., 2016). Un claro exponente de ello es la creación de INTERNIC (INTERNet Network Information Center), organismo de administración de Internet creado por la Fundación Nacional para la Ciencia (por sus siglas en inglés, NSF) (Riemann, 2012). Con el transcurso del tiempo se ha extendido el acceso de Internet en casi todas las regiones del planeta posibilitando que gran parte de la población mundial acceda de forma rápida y segura a un gran cúmulo de información (Filippi, 2009) como por ejemplo, Cuba que actualmente realiza grandes esfuerzos para acceder a la red de redes.

A medida que el país ha mejorado su conectividad global, se ha apreciado el impacto que esta tiene en el área empresarial, debido a que ha eliminado las barreras que antes existían para socializar la información. Esto nos plantea un escenario en el que las empresas necesitan dar a conocer su catálogo de servicios y de productos a través de internet para poder satisfacer las necesidades de los clientes, que a diferencia de las generaciones antecedentes cuentan con un nivel de información superior, dando lugar de esta forma a los sitios web (Nacional, 2015).

Los sitios Web deben brindar una imagen de calidad y constante progreso, por lo que es de vital importancia que permitan publicar, gestionar y organizar una amplia variedad de contenido de forma ágil y sencilla, por lo que se debe considerar para su desarrollo los Sistema de Gestión de Contenidos (por sus siglas en inglés, CMS), pues permiten una fácil administración y publicación de los contenidos (Colombia, 2016). Según un estudio de la Universidad de Málaga de España, realizado en el 2013, entre los gestores de contenidos utilizados para el desarrollo de portales web desde el año 2004 hasta la actualidad a nivel mundial, se encuentra Drupal (Consulting, 2013).

La Universidad de Ciencias Informática (UCI) es una institución destinada a producir aplicaciones y servicios informáticos, a partir de la vinculación estudio-trabajo como modelo de formación y servir de soporte a la industria cubana de la informática (www.uci.cu, 2017). Uno de los centros destinados a la producción de software es el Centro de Ideoinformática (CIDI), perteneciente a la Facultad 1. Entre las líneas de investigación y desarrollo del centro se encuentra la informatización de la sociedad cubana y una de las premisas es el desarrollo de portales web institucionales, periódicos digitales y portales empresariales en Internet. En el centro se utiliza el CMS Drupal para el desarrollo de los sitios, pues permite gestionar los contenidos desde el propio portal, mediante formularios que facilitan su gestión a

usuarios con pocos conocimientos del desarrollo web, sin embargo; en situaciones donde el acceso a Internet es limitado, se dificulta el proceso de gestión de los contenidos. Para mejorar esta situación se decidió agregar una nueva funcionalidad al portal de CubaSí que permitiera a las personas con permisos para publicar contenidos web en el sitio hacerlo vía correo electrónico. Con el propósito de implementar esta funcionalidad se integró el módulo Mailhandler obtenido del sitio Drupal.org que actualmente es la web oficial de Drupal, pero no se han obtenido los resultados esperados pues se han presentado deficiencias tales como:

- Escasa documentación de cómo configurar el módulo para su correcto funcionamiento impidiendo su correcto uso por los desarrolladores del sitio.
- Presencia de poca fiabilidad de los contenidos publicados pues una vez creadas correctamente las cuentas a cada usuario del sistema, los correos electrónicos son publicados automáticamente en el sitio, sin hacer un previo análisis del contenido web que se publicará.
- Imposibilidad de configurar una cuenta predeterminada que permita a los usuarios autenticados en el portal publicar contenido web a través de una única dirección de correo electrónico.
- Dificultad para publicar archivos multimedia a través de los correos electrónicos.

Debido a la situación problemática anteriormente expuesta, se define el siguiente **problema de investigación**:

¿Cómo contribuir a la publicación de contenidos web vía correo electrónico en el portal de CubaSí desarrollado en Drupal?

Se enmarca el **objeto de estudio** en el proceso de publicación de contenidos web usando el CMS Drupal. Teniendo en cuenta el problema de investigación, se establece como **objetivo general** de la investigación, desarrollar un módulo que contribuya a la publicación de contenidos web vía correo electrónico para el portal de CubaSí desarrollado en Drupal.

Delimitando así el **campo de acción** al proceso de publicación de contenidos web vía correo electrónico en el portal de CubaSí desarrollado en Drupal.

Objetivos específicos

- Construir el marco teórico conceptual de la investigación.

- Realizar un estudio de los sistemas homólogos y algunas de las tecnologías empleadas para el desarrollo de un módulo para la publicación de contenidos web vía correo electrónico para el portal de CubaSí desarrollado en Drupal.
- Definir los requisitos funcionales y no funcionales del módulo para la publicación de contenidos web vía correo electrónico para el portal de CubaSí desarrollado en Drupal.
- Implementar las funcionalidades identificadas para desarrollar un módulo para la publicación de contenidos web vía correo electrónico para el portal de CubaSí desarrollado en Drupal.
- Validar las funcionalidades del módulo para la publicación de contenidos web vía correo electrónico para el portal de CubaSí desarrollado en Drupal.

Se formula la siguiente **hipótesis de investigación**: El módulo para la publicación de contenido web vía correo electrónico contribuirá a la mejora del proceso de publicación de contenidos en el portal de CubaSí desarrollado en Drupal.

En el desarrollo de la hipótesis se define como **variable dependiente**: Contribución a la mejora del proceso de publicación de contenido web en el portal de CubaSí, dicha variable hace alusión al aprovechamiento de la tecnología por parte de los usuarios autenticados en el portal por la usabilidad, seguridad y accesibilidad que el mismo brinda.

Como **variable independiente** se define: Módulo para la publicación de contenido web vía correo electrónico, el mismo consiste en un módulo desarrollado para Drupal con un conjunto de funcionalidades que permiten publicar contenido web vía correo electrónico en el portal de CubaSí.

Tabla 1. Operacionalización de las variables

Var iab les	Dimensiones	Indicadores	Medidas	Unidad de medida
VD	Aprovechamiento de la tecnología	Usabilidad	Estructura simple	Muy Adecuada > 50% Bastante Adecuada > 30% 16% <Adecuada<29% 8% < Poco Adecuada < 15%

				Inadecuada < 8%
			Interfaz intuitiva	Muy Adecuada > 50% Bastante Adecuada > 30% 16% <Adecuada<29% 8% < Poco Adecuada < 15% Inadecuada < 8%
			Navegabilidad	Muy Adecuada > 50% Bastante Adecuada > 30% 16% <Adecuada<29% 8% < Poco Adecuada < 15% Inadecuada < 8%
		Seguridad	Confidencialidad	Satisfactoria Insatisfactoria
			Integridad	Satisfactoria Insatisfactoria
		Rendimiento	Tiempo de respuesta	Satisfactoria Insatisfactoria
VI	Nivel tecnológico	Nuevas funcionalidades	Funcionalidad que permita publicar en el portal de CubaSí vía correo electrónico	Muy Adecuada > 50% Bastante Adecuada > 30% 16% <Adecuada<29% 8% < Poco Adecuada < 15% Inadecuada < 8%
			Funcionalidad que	Muy Adecuada > 50%

			<p>permita crear cuentas desde las cuales los usuarios autenticados puedan publicar en el sitio desde una única dirección de correo electrónico.</p>	<p>Bastante Adecuada > 30% 16% <Adecuada<29% 8% < Poco Adecuada < 15% Inadecuada < 8%</p>
		Integración con CubaSí	<p>El módulo al ser integrado con el portal de CubaSí no afecta el correcto funcionamiento del mismo, y viceversa.</p>	<p>Satisfactoria Insatisfactoria</p>
		Conexión con el servidor de correo	<p>Accesibilidad a las cuentas de correo autenticadas en el módulo.</p>	<p>Muy Adecuada > 50% Bastante Adecuada > 30% 16% <Adecuada<29% 8% < Poco Adecuada < 15% Inadecuada < 8%</p>
			<p>Acceso del módulo a los campos del correo electrónico para su posterior publicación.</p>	<p>Muy Adecuada > 50% Bastante Adecuada > 30% 16% <Adecuada<29% 8% < Poco Adecuada < 15% Inadecuada < 8%</p>

Tareas de la Investigación:

- Sistematización sobre el estudio de sistemas homólogos.
- Definición de los conceptos asociados al módulo para la creación de contenidos web vía correo electrónico.
- Selección de las tecnologías, herramientas y estándares que se necesitan para implementar la propuesta de solución.
- Identificación de la metodología que guiará el proceso de desarrollo de software.
- Definición de los requisitos funcionales y no funcionales de la propuesta de solución.
- Análisis y diseño de la propuesta de solución.
- Implementación de la propuesta de solución.
- Realización de las pruebas para validar las funcionalidades de la propuesta de solución.

Métodos Teóricos

Analítico-Sintético: La utilización de este método posibilitó el estudio de las fuentes bibliográficas y de los conceptos fundamentales sobre la integración del módulo para la publicación de contenidos web vía correo electrónico, lo que permitió determinar las principales características que debe tener la propuesta de solución.

Análisis Histórico-Lógico: Se utilizó para analizar la evolución y la trayectoria histórica de soluciones informáticas relacionadas con el objeto de estudio de la investigación, así como las herramientas, tecnologías y estándares utilizados para su desarrollo.

Modelación: Utilizado en la representación, mediante el uso de diagramas, de las características del sistema, relaciones entre objetos y componentes.

Métodos Empíricos

Observación: Este método permitió obtener la información necesaria sobre los módulos para la publicación de contenidos web vía correo electrónico, el cual arrojó como resultado la definición de las principales funcionalidades y características que pueden ser utilizadas en la propuesta de solución.

Encuesta: Este método se empleó con el objetivo de recopilar información del estado de opinión de los expertos seleccionados, con respecto a los indicadores a evaluar del módulo que conformará la propuesta de solución.

El presente documento se estructura de la siguiente forma:

Capítulo I. Fundamentación teórica: Se definen y describen las herramientas, tecnologías y metodologías necesarias para desarrollar la propuesta de solución, además se analiza el estado del arte de sistemas homólogos.

Capítulo II. Análisis y diseño del Módulo para la publicación de contenidos web vía correo electrónico: En este capítulo se describe la propuesta de solución para el problema de investigación anteriormente definido. Se definen los requisitos funcionales y no funcionales del sistema y se realiza el análisis y diseño del módulo para la publicación de contenidos web vía correo electrónico para el portal de CubaSí desarrollado en Drupal.

Capítulo III. Implementación y validación del Módulo para la publicación de contenidos web vía correo electrónico: Se describe la propuesta de solución a través de un conjunto de artefactos y diagramas generados en el sistema. También se describen los tipos de pruebas realizadas a la solución propuesta para comprobar su correcto funcionamiento presentando y se describen detalladamente los resultados obtenidos de ellas.

Capítulo I: Fundamentación teórica

1.1. Introducción

En el capítulo se define la base teórica y conceptual para el desarrollo de la propuesta de solución, se abordan los conceptos relacionados con el proceso de publicación de contenidos web vía correo electrónico. También se realiza un estudio del estado del arte de los sistemas homólogos. Se analizan las tecnologías, metodologías y herramientas más factibles para utilizar en el proceso de desarrollo.

1.2. Conceptos asociados al módulo para la publicación de contenidos web vía correo electrónico

En esta sección se explican algunos aspectos fundamentales para la comprensión del contexto de la investigación.

Módulo

Según la Real Academia Española es una pieza o conjunto unitario de piezas que se repiten en una construcción de cualquier tipo, para hacerla más fácil, regular y económica (Española, 2017).

En programación, un módulo es un software que agrupa un conjunto de subprogramas y estructuras de datos. Los módulos son unidades que pueden ser compiladas por separado y los hace reusables y permite que múltiples programadores trabajen en diferentes módulos de forma simultánea, minimizando los tiempos de desarrollo. Los módulos promueven la modularidad y el encapsulamiento, permitiendo generar programas complejos de fácil comprensión (Alegsa, 2010).

Publicación

Según el Diccionario de la lengua española es una obra literaria o artística publicada, difusión o comunicación de cualquier información para que sea conocida, difusión de algo por medio de la imprenta o cualquier otro procedimiento técnico (española, 2005).

El término publicación se utiliza para hacer referencia al acto mediante el cual se hace público o se da a conocer determinada información, acto, dato, etc (ABC, 2017).

Contenido web

Unidades de información que abarcan una temática homogénea, que pueden diferenciarse con cierta precisión y que suelen tener un mantenimiento simultáneo. La elaboración y propuesta de contenidos tiene siempre la visión de ofertar a clientes informaciones útiles para la toma de decisiones, con lo que, en última instancia, estamos hablando de un servicio público (Jaurilaritza, 2012).

Correo electrónico

Según la real academia es sistema de transmisión de mensajes o archivos de un terminal a otro a través de redes informáticas (Española, 2015).

El correo electrónico es una aplicación de Internet que permite el intercambio de archivos entre diferentes ordenadores conectados a la red. Los servicios de correo electrónico otorgan a sus usuarios un buzón electrónico, es decir, un espacio que se identifica a través de una secuencia única e irrepetible que constituye la dirección electrónica (Martínez, 2015).

Desde un enfoque lingüístico, el correo electrónico se define como un sistema de interacción mediatizada que soporta intercambios de tipo individual o colectivo, real o virtual y que se sustenta en la transacción de textos digitalizados con una forma, función, estructura, lengua y estilo propios (Hidalgo, 2013).

1.3. Sistemas homólogos

A continuación, se analizan soluciones informáticas existentes para la publicación de contenidos web vía correo electrónico en los CMS, con el objetivo de establecer aspectos comunes o elementos que se deban mejorar o integrar a la propuesta de solución que se desarrollará.

1.3.1. Mailhandler

Mailhandler es un módulo desarrollado para el CMS Drupal que permite la publicación de contenidos web vía correo electrónico. Está disponible en la página web oficial del CMS Drupal. Para su correcto funcionamiento debe ser configurado un nuevo correo electrónico desde el cual se enviará el contenido que se utilizará en la publicación, además se debe instalar y configurar los módulos Feeds y CTool.

Se deben establecer los siguientes parámetros en el módulo (Raul, 2016):

- Protocolo: IMAP
- Carpeta: *INBOX*
- Dominio: imap.gmail.com
- Puerto: 993
- Usuario: username@gmail.com (ejemplo)
- Contraseña: mypassword (ejemplo)
- Comandos extras: /ssl

Deficiencias que puede presentar el módulo Mailhandler (Raul, 2016):

- El servidor puede presentar problemas al no aceptar los siguientes elementos:
 1. URLs de fácil comprensión.
 2. Protocolo SSL o acceso al servidor de correo a través del puerto 993.
 3. Límite de conexiones SQL.
- Pueden afrontar problemas con la resolución de dominios. Esta característica se verifica primeramente haciendo un ping #ping midominio.org, si el resultado es negativo el usuario debe comunicarse con el proveedor de nombres de dominio.

1.3.2. JetPack

JetPack es un *plugin* para WordPress, es una extensión de las funcionalidades básicas de este CMS. Está formado por gran cantidad de módulos que se pueden activar o desactivar dependiendo de las necesidades del sitio que está siendo creado. Todos sus módulos son completamente gratuitos y puede ser descargado desde el repositorio oficial de WordPress (Networks, 2017).

Entre las funcionalidades de este *plugin* se encuentra la de “Publicar por correo electrónico” que como su nombre indica, permite publicar en el blog enviando el artículo en una determinada sintaxis a un servicio que se encargará de procesar el envío y publicar el artículo en WordPress. Al activar esta funcionalidad se puede crear una dirección de correo electrónico única, a la que cada usuario desde su correo electrónico personal enviará un mensaje para que sea publicado en el sitio (Hostdime, 2014).

Otras características importantes son (Hostdime, 2014):

- Los usuarios con perfil de colaboradores no podrán publicar mensajes inmediatamente, el mensaje que envían estará pendiente de revisión por el administrador del sitio web.
- Se pueden especificar una o varias categorías para el artículo. Si no existe la categoría, esta se creará automáticamente.
- Posibilita configurar la contraseña del artículo, en caso que sea privado.

Esta funcionalidad presenta varias deficiencias que limitan las publicaciones de contenidos vía correo, pues no permite publicar en los artículos imágenes, videos, asignar imagen destacada, incluso enviar el artículo como correo.

1.3.3. MMS blogs

Este componente para Joomla permite publicar desde un correo electrónico un nuevo artículo en el sitio web desde un correo electrónico. Dispone de una función que permite recuperar los mensajes de las cuentas de correo a intervalos regulares (por ejemplo, 15 minutos) para que se visualice como un blog.

Permite elegir secciones, categorías y los derechos de publicación. Los mensajes MMS o correos electrónicos pueden estar acompañados por archivos multimedia de imágenes y videos (Lorenzo, 2012).

Entre las funcionalidades principales de MMS blogs se encuentran (Lorenzo, 2012):

- El envío de notificaciones de la recepción de mensajes por correo electrónico a los redactores del sitio desarrollado en Joomla.
- Administración de varios editores con una recepción y publicación de una o varias direcciones de correo electrónico.
- Gestiona las categorías y las secciones.
- Permite ver la lista de categorías directamente desde el panel, lo que posibilita que una o más direcciones se puedan vincular a una categoría específica, es decir cada usuario podrá escribir a una dirección de correo electrónico en relación con el tema de su mensaje.

Para acceder a los servicios de MMS blogs la versión php del servidor debe tener la extensión "php_imap" instalada y activada. Este requisito limita el uso de este módulo, pues el usuario debe instalar esta librería en el servidor web que utilice, para poder acceder al servidor de correo (Lorenzo, 2012).

1.3.4. Resultados del análisis de los sistemas homólogos

Luego de realizar un estudio a soluciones informáticas existentes para la publicación de contenidos web vía correo electrónico en los CMS, se llegó a las siguientes conclusiones:

1. Los módulos anteriormente analizados no responden a las necesidades de desarrollo del centro CIDI como, por ejemplo:
 - MMS blogs y JetPack son componentes para ser usados en Joomla y en WordPress respectivamente, siendo así incompatibles con Drupal que es el CMS en el cual se ha desarrollado el portal de CubaSí.
 - Mailhandler no permite publicar archivos multimedia a través de los correos electrónicos, ni crear una cuenta predeterminada desde la cual los usuarios autenticados en el sitio web puedan publicar contenido web desde una dirección de correo electrónico única.
2. Desarrollar un módulo para la publicación de contenidos web vía correo electrónico que responda a las necesidades del centro CIDI.
3. Para el desarrollo de la propuesta de solución se deben tener en cuenta las características fundamentales de los sistemas homólogos anteriormente analizados, tales como:

- Funcionalidad que permita la publicar contenidos web vía correo electrónico en formato HTML.
- Adaptabilidad del módulo en diferentes entornos de desarrollo.
- Reutilización de su código fuente.
- Posibilidad de publicar archivos multimedia de imágenes y documentos.
- Permitir la configuración de más de una dirección de correo electrónico única para que todos los usuarios con permisos para publicar contenido web en el portal de CubaSí, puedan acceder a esta funcionalidad a través de su correo electrónico personal.
- Interfaz simple e intuitiva.
- Fácil configuración del módulo para clientes con pocos conocimientos informáticos

En la Tabla 2 se representa la comparación de las características y funcionalidades principales que presentan el módulo Mailhandler, el *plugin* JetPack y el componente MMS blogs.

Tabla 2. Comparación de los sistemas homólogos

Sistemas Homólogos	Compatibilidad con Drupal	Fácil Configuración	Múltiples usuarios	Publicación Automática	Publicación de archivos multimedia	Interfaz de estructura simple	Cuenta Predeterminada
MailHandler	Sí	No	Sí	Sí	No	No	No
JetPack	No	Sí	Sí	No	No	Sí	Sí
MMS blogs	No	Sí	Sí	No	Sí	Sí	Sí

1.4. Metodología para el desarrollo de software

Una metodología de desarrollo de software describe un entorno que es usado para organizar, planificar, y dirigir un proceso de desarrollo de un software. Existen varias metodologías de desarrollo de software, todas contienen algunas etapas básicas del ciclo de desarrollo de software como son la planificación, análisis, diseño, implementación y mantenimiento (Verma, y otros, 2014). Las metodologías se pueden agrupar en dos grandes grupos, las tradicionales y las ágiles.

1.4.1. Metodologías Ágiles

Las metodologías tradicionales se centran especialmente en el control del proceso, estableciendo rigurosamente las actividades involucradas, los artefactos que se deben producir, además de las herramientas y notaciones que se usarán (Canós, 2014).

Las metodologías ágiles dan mayor valor al individuo, a la colaboración con el cliente y al desarrollo incremental del software con iteraciones muy cortas. En estas el cliente llega a formar parte del equipo de trabajo (Canós, 2014).

En la investigación se decide utilizar una metodología ágil porque son dinámicas y muy flexibles permitiendo cambios durante el proyecto, también permite generar pocos artefactos. Además, los grupos de desarrollo dentro de las metodologías ágiles son pequeños y trabajan en el mismo sitio, esta característica define al grupo de desarrollo de la investigación y del Departamento de Servicios Informáticos para Internet (SENIT) del Centro Ideoinformática (CIDI), donde actualmente se desarrolla el portal de CubaSí. Por otra parte, las metodologías tradicionales tienen grupos de desarrollo más grandes y se encuentran distribuidos, son estrictas y con cierta resistencia a cambios y generan muchos artefactos.

AUP-UCI

Para el proceso de desarrollo de software es necesario contar con una metodología que lo guíe. En la Universidad de Ciencias Informáticas (UCI) se decidió realizar una variación de la metodología del Proceso Unificado Ágil (por sus siglas en inglés, AUP), de forma tal que se adapte al ciclo de vida definido para la actividad productiva de la universidad (Sánchez, 2015).

Para aplicar buenas prácticas AUP-UCI se apoya en el Modelo CMMI-DEV v1.3, el cual constituye una guía para aplicar las mejores prácticas en una entidad desarrolladora. Estas prácticas se centran en el desarrollo de productos y servicios de calidad (Sánchez, 2015).

De las 4 fases que propone AUP (Inicio, Elaboración, Construcción, Transición) se decide para el ciclo de vida de los proyectos de la UCI mantener la fase de Inicio, pero modificando el objetivo de la misma, se unifican las restantes 3 fases de AUP en una sola, a la que se le llama Ejecución y se agrega la fase de Cierre (Sánchez, 2015).

Se decide utilizar para el desarrollo de esta investigación la metodología ágil AUP-UCI pues es la metodología empleada en el Departamento de Servicios Informáticos para Internet (SENIT) del Centro Ideoinformática (CIDI), el cual es un proyecto real donde actualmente se desarrollan sitios web como, por

ejemplo, el portal de CubaSí. Permite aplicar buenas prácticas pues se apoya en el Modelo CMMI-DEV v1.3. Esta metodología fue seleccionada por las ventajas que presenta al permitir la interrelación y el aporte de los implicados sobre los entregables, además está diseñada para equipos de trabajos pequeños y posibilita detectar errores tempranos a través de un ciclo iterativo.

1.5. Sistema de gestión de contenidos

Un Sistema de Gestión de Contenidos (por sus siglas en inglés, CMS) es una aplicación de web que se utiliza para crear, editar, administrar y publicar contenido de forma coherente y eficaz. Los CMS son capaces de almacenar datos, control de versiones, la búsqueda, así como la documentación específica de la recuperación y el control de la industria y la administración de archivos incluyendo noticias, artículos, informes manuales, folletos de marketing entre otros (Bruno, 2014).

Los CMS permiten a cualquier usuario sin conocimientos técnicos administrar fácilmente y rápidamente su página web. El uso de un gestor de contenidos ofrece a los propietarios de sitios web la posibilidad de añadir tanto contenido como deseen y administrar su contenido en general de manera eficiente y rentable (Bruno, 2014).

Drupal 7

El portal de CubaSí fue desarrollado en Drupal por lo que se decide utilizar este CMS para integrar la propuesta de solución. Drupal 7 es de código abierto, altamente modular con énfasis en la colaboración. Contiene funcionalidades básicas en su núcleo y se puede agregar funcionalidades a través de la instalación de módulos. Separa con éxito la gestión de contenido de la presentación de contenidos. Es distribuido bajo los términos de la Licencia Pública General (por sus siglas en inglés, GNU/GPL). Drupal es un marco de gestión de contenido (CMF). Proporcionar herramientas para la construcción de sitios web, ofrece vías para que programadores y desarrolladores personalicen Drupal usando módulos. También cuenta con una gran comunidad de desarrolladores que trabajan constantemente en la actualización y modernización del sistema (Hispano, 2011).

Características principales

- Sistema de módulos flexible: Los módulos son una característica de Drupal que pueden modificar y agregar funcionalidades en un sitio web.
- Sistema de temas personalizable: Las salidas en Drupal son totalmente personalizables.

- Sistema de contenido extensible y de entidades: Se pueden definir nuevos tipos de contenido (blogs, eventos, noticias) y agregar campos para distintos tipos de contenido.
- Permisos de acceso basado en Roles: Cada usuario en Drupal contiene un rol con privilegios determinados.
- Herramientas para publicaciones y colaboraciones: Drupal contiene soportes para herramientas como comentarios, fórums, y perfiles de usuarios personalizados.

Ventajas

- Drupal es un software de código abierto que cuenta con una amplia comunidad de usuarios y desarrolladores.
- Soporta más de 55 idiomas.
- El sitio oficial de Drupal lista alrededor de 12101 módulos y temas gratuitos para extender sus funcionalidades.

1.6. Herramientas para el desarrollo

A continuación, se definen y describen las herramientas y tecnologías a utilizar para el desarrollo de la propuesta de solución.

1.6.1. Lenguaje de modelado

UML 2.5

El Lenguaje de Modelado Unificado (por sus siglas en inglés, UML) es la sucesión de una serie de métodos de análisis y diseño orientadas a objetos que aparecen a fines de los años 80 y principios de los 90. UML es llamado un lenguaje de modelado, no un método. Los métodos consisten de ambos de un lenguaje de modelado y de un proceso. El lenguaje de modelado es la notación (principalmente gráfica) que usan los métodos para expresar un diseño. El proceso indica los pasos que se deben seguir para llegar a un diseño (Cornejo, 2008).

1.6.2. Lenguaje de programación

PHP 5

El pre-procesamiento de hipertexto (por sus siglas en inglés, PHP), es un lenguaje de código abierto muy usado, especialmente adecuado para desarrollo web y que puede ser incrustado en HTML. Este lenguaje tiene características que lo convierten en la herramienta ideal para la creación de páginas web dinámicas (PHP, 2012):

- Soporte para bases de datos: MySQL, PostgreSQL, Oracle, MS SQL Server, SybasemSQL, Informix, entre otras.
- Integración con varias bibliotecas externas, permite generar documentos en formato PDF hasta analizar código XML.
- Ofrece una solución simple y universal para las paginaciones dinámicas de la web de fácil programación.
- Soportado por una gran comunidad de desarrolladores. Como producto de código abierto, PHP goza de la ayuda de un gran grupo de programadores, permitiendo que los fallos de funcionamiento se encuentren y reparen rápidamente.
- El código es actualizado continuamente con mejoras y extensiones de lenguaje para ampliar las capacidades de PHP.
- Bases de datos locales: permite el uso de una base de datos local, con la que se puede trabajar en una página web por medio del cliente y a través de un API.

CSS 3

Las Hojas de Estilo en Cascada (por sus siglas en inglés, CCS) es un lenguaje utilizado para describir el aspecto y el formato de los documentos HTML. CSS ha sido ampliamente adoptado en la web y la práctica del desarrollo móvil, ya que permite una separación limpia del contenido de la presentación. La separación de los contenidos y su presentación presenta numerosas ventajas, ya que obliga a crear documentos HTML/XHTML bien definidos y con significado completo (también llamados “documentos semánticos”. Esta separación mejora la accesibilidad del documento, reduce la complejidad de su mantenimiento y permite visualizar el mismo documento en infinidad de dispositivos diferentes (Paganotti, 2013).

Las nuevas características de CSS3 respecto a sus versiones anteriores incluyen soporte para selectores adicionales, sombras, esquinas redondeadas, múltiples fondos, animaciones y transparencia.

HTML 5

El lenguaje de marcado de hipertexto (por sus siglas en inglés, HTML), es un lenguaje de publicación especificado como un estándar por el W3C (World Wide Web Consortium) que permite la creación de páginas web. Inicialmente fue presentado por Tim Berners-Lee que propuso un sistema basado en hipertexto como resultado de la aplicación de SGML para el intercambio de información en la web. La

aparición del lenguaje influyó notablemente en el crecimiento de Internet, dónde la información era distribuida mediante colecciones fragmentadas de textos, imágenes y sonidos. HTML es independiente de la plataforma utilizada y se basa fundamentalmente en el uso de etiquetas estructurales y semánticas, adecuadas para la creación de documentos relativamente simples que permiten simplificar su estructura (Bernardo, 2012).

1.6.3. Herramientas CASE

Se puede definir a las herramientas de Ingeniería de Software Asistida por Ordenador (por sus siglas en inglés, CASE) como una herramienta que se utiliza para automatizar alguna actividad.

Existen variadas herramientas CASE ayudan en las tareas relacionadas con la fase de desarrollo del software como la especificación, estructurado, análisis, diseño, codificación, pruebas y otras actividades como gestión de proyectos y gestión de la configuración (Bareisa, 2015).

Visual Paradigm 8.0

Para el modelado del proceso se utilizará la herramienta Visual Parading, pues soporta el ciclo de vida completo del desarrollo de software: análisis y diseño orientados a objetos, implementación y pruebas. Permite construir diagramas de diversos tipos, código inverso, generar código desde diagramas y generar documentación. Esta herramienta proporciona abundantes tutoriales de UML, demostraciones interactivas de UML y proyectos UML (Parading, 2015).

Se seleccionó esta herramienta pues en comparación con otras como Microsoft Project, CaliberRM® y Rational Rose tiene una licencia gratuita y comercial, es fácil de instalar y actualizar y compatible entre ediciones.

1.6.4. Entorno integrado de desarrollo

Un entorno de desarrollo integrado (por sus siglas en inglés, IDE), es un entorno de programación que ha sido empaquetado como un programa de aplicación, es decir, consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica (por sus siglas en inglés, GUI) (García, 2013).

Los IDE ofrecen un marco de trabajo para la mayoría de los lenguajes de programación tales como C++, Python, Java, C#, Delphi, Visual Basic, etc. En algunos lenguajes, un IDE puede funcionar como un sistema en tiempo de ejecución, en donde se permite utilizar el lenguaje de programación en forma interactiva, sin necesidad de trabajo orientado a archivos de texto (García, 2013).

NetBeans 8.0

En el desarrollo de la propuesta de solución se utilizará el IDE NetBeans pues es de código abierto y posee una plataforma de aplicación, la cual puede ser usada como una estructura de soporte general para compilar cualquier tipo de aplicación. Netbeans tienen la ventaja de ser un producto libre y gratuito sin restricciones de uso, por lo que no se necesitaran pagar licencias por desarrollar aplicaciones con el uso de este. El uso de Netbeans tiene como resultado un código simple, fácil de mantener y legible (para los diseñadores de páginas) (Cahueñas, y otros, 2011).

La plataforma ofrece servicios comunes a las aplicaciones de escritorio, permitiéndole al desarrollador enfocarse en la lógica específica de su aplicación. Entre las características de la plataforma están (Oracle, 2014):

- Administración de las interfaces de usuario (ej. menús y barras de herramientas).
- Administración de las configuraciones del usuario.
- Administración del almacenamiento (guardando y cargando cualquier tipo de dato).
- Administración de ventanas.

Entre sus principales ventajas se encuentra:

- La plataforma Netbeans puede ser usada para desarrollar cualquier tipo de aplicación.
- Reutilización de Módulos.
- Permite el uso de la herramienta Update Center Module.
- Instalación y actualización simple.
- Incluye *Templates* y *Wizards*.
- Posee soporte para PHP.

1.6.5. Servidor web

La principal tarea de un servidor web es alojar sitios y/o aplicaciones, las cuales son accedidas por los clientes utilizando un navegador que se comunica con el servidor mediante el protocolo HTTP. El servidor espera peticiones del cliente y le da respuesta con el contenido solicitado. Disponen de intérpretes de lenguajes de programación que ejecutan códigos embebidos dentro del HTML de las páginas que contiene el sitio antes de enviar el resultado al cliente (Morales , et al., 2009).

Apache 2.4

Se decide utilizar el servidor web Apache en su versión 2.4 pues a diferencia de NGINX y Lighttpd (Vojacek, 2012):

- Provee la información de manera sincrónica, es decir, mantiene una copia de cada solicitud realizada y trata de proveerla acorde a cada pedido, para los otros 2 servidores esto se realiza de manera asincrónica y por pedido, esto quiere decir que NGINX por ejemplo solo entrega lo que es solicitado, no prepara los recursos para entregar la información como lo hace Apache.
- NGINX y Lighttpd presentan como deficiencias la falta de módulos, pues en comparación con ellos Apache provee cientos de módulos para poder configurar e instalar los “Apache Modules”, que no solo se instalan, sino que se pueden activar/desactivar de manera muy sencilla y permiten realizar funciones adicionales.

Apache 2.4 presenta mejoras en el rendimiento, que se sintetizan en un menor consumo de memoria y mejoras en la concurrencia de las peticiones. Los diferentes módulos de multiprocesado (MPM) disponibles en Apache 2.4 permiten a los administradores de sistemas ajustar Apache para ser más rápido según las necesidades y la naturaleza de las peticiones que tenga que atender. Estos módulos pueden ser seleccionados en tiempo de ejecución con lo que añade una mayor flexibilidad (Rodríguez, 2012).

Entre las novedades más destacadas de Apache 2.4 según (Rodríguez, 2012) se encuentran:

- Rendimiento mejorado (menor consumo de recursos y mejor tratamiento de la concurrencia)
- Menor uso de memoria
- Soporte para entrada/salida asíncrona
- Configuración de proxy dinámico inverso (Dynamic reverse proxy)
- Soporte detallado del uso de caché y *proxie* alto nivel de tráfico.

1.6.6. Sistema gestor de bases de datos

Definimos un Sistema Gestor de Bases de Datos o SGBD (por sus siglas en inglés, DBMS) como una colección de datos relacionados entre sí, estructurados y organizados, y un conjunto de programas que acceden y gestionan esos datos. La colección de esos datos se denomina Base de Datos o BD, (por sus siglas en inglés, DB) (Martín, 2007).

Una BD es un gran almacén de datos que se define una sola vez; los datos pueden ser accedidos de forma simultánea por varios usuarios; están relacionados y existe un número mínimo de duplicidad;

además en las BBDD se almacenarán las descripciones de esos datos, lo que se llama metadatos en el diccionario de datos (Martín, 2007).

El SGBD es una aplicación que permite a los usuarios definir, crear y mantener la BD y proporciona un acceso controlado a la misma. Debe prestar los siguientes servicios (Martín, 2007):

- Creación y definición de la BD: especificación de la estructura, el tipo de los datos, las restricciones y relaciones entre ellos mediante lenguajes de definición de datos.
- Toda esta información se almacena en el diccionario de datos, el SGBD proporcionará mecanismos para la gestión del diccionario de datos.
- Manipulación de los datos realizando consultas, inserciones y actualizaciones de los mismos utilizando lenguajes de manipulación de datos.
- Acceso controlado a los datos de la BD mediante mecanismos de seguridad de acceso a los usuarios.
- Mantener la integridad y consistencia de los datos utilizando mecanismos para evitar que los datos sean perjudicados por cambios no autorizados.
- Acceso compartido a la BD, controlando la interacción entre usuarios concurrentes.

MySQL 5

Como SGBD se empleará MySQL 5 el cual utiliza el lenguaje de consulta estructurado (por sus siglas en inglés, SQL). Es un producto de código abierto desarrollado bajo la Licencia Pública General (por sus siglas en inglés, GNU), lo que hace posible que pueda ser modificado por los usuarios que lo utilizan. Es un SGBD usado y estandarizado para acceder a base de datos relacionales. Facilita la integración con programas desarrollados en C y C++ pues fue desarrollado en este lenguaje. Es un sistema multiplataforma, totalmente funcional en sistemas operativos como Linux, MacX, UNIX y Microsoft Windows. Permite la creación de base de datos con acceso desde páginas web dinámicas, dando la posibilidad de realizar múltiples y rápidas consultas (Thibaud, 2006).

1.6.7. Herramientas de prueba

Para validar las aplicaciones y sitios web se utilizan herramientas de prueba que permiten comprobar el rendimiento general y el cumplimiento de las especificaciones del sistema al ser sometidos a diferentes pruebas. Una prueba tiene éxito si descubre un error no detectado hasta entonces (Murcia, 2009).

Acunetix Web Vulnerability Scanner

Acunetix Web Vulnerability rastrea el sitio web, analiza automáticamente las aplicaciones web y encuentra vulnerabilidades por inyección de SQL, *cross-site scripting* y otras vulnerabilidades que exponen el negocio en línea. Informes concisos permiten identificar en qué punto necesitan ser parcheadas sus aplicaciones Web y permitiendo por tanto proteger su negocio de inminentes ataques de piratas informáticos (Compunet, 2016).

Acunetix ha sido pionero en la tecnología de análisis de seguridad de aplicaciones web y ha establecido una ventaja de ingeniería en el análisis web y detección de vulnerabilidades con las siguientes características innovadoras (Compunet, 2016):

- AcuSensor La tecnología permite una lectura precisa con pocos positivos falsos, combinando técnicas de exploración de caja negra, con la retroalimentación de sus sensores colocados en el interior de la fuente código.
- Un analizador automático JavaScript para pruebas de seguridad de AJAX y aplicaciones Web 2.0.
- La prueba más avanzada y profunda de la industria para Inyección SQL y Cross-Site Scripting.
- El grabador de macros hace que las pruebas sobre los formularios de las áreas protegidas de la Web sean más fáciles.
- Escáner rápido capaz de rastrear cientos de miles de páginas sin interrupciones
- Pruebas de vulnerabilidad de carga automatizada de informes.
- AcunetixDeepScan entiende tecnologías web complejas como SOAP, XML, AJAX y JSON.

Apache Jmeter

Apache JMeter es una de las herramientas Open Source para realizar pruebas de carga. Además de HTTP soporta protocolos como POP3, IMAP, FTP, entre otros y permite programar lo que llama planes de prueba que consisten en guiones de ejecución para las pruebas de carga. Es un software de código abierto, una aplicación diseñada para cargar la conducta funcional de prueba y medida de rendimiento. (Apache, 2016).

Jmeter puede ser utilizado para probar el rendimiento tanto en los recursos estáticos y dinámicos de un sitio web. Se puede utilizar para simular una carga pesada en un servidor, grupo de servidores, la red o el objeto a probar su resistencia o para analizar el rendimiento general bajo diferentes tipos de carga. (Apache, 2016).

1.7. Conclusiones parciales

Del estudio realizado en el capítulo se puede concluir lo siguiente:

- El estudio del estado del arte de los sistemas homólogos existentes, evidenció la presencia de deficiencias en su funcionamiento y que no cumplen con de las necesidades de desarrollo del centro CIDI, por lo que se concluye que no pueden formar parte de la propuesta de solución, pero sí se deben tener en cuentas algunas de sus funcionalidades y la mejora de sus limitaciones, para el desarrollo de la propuesta de solución.
- La selección y estudio de la base tecnológica para el desarrollo de la propuesta de solución permitió definir AUP-UCI como metodología de desarrollo de software para guiar este proceso.
- Se seleccionaron como herramientas a utilizar para el desarrollo de la propuesta de solución NetBeans como entorno de desarrollo integrado, Apache 2.4 como servidor web, MySQL 5 como SGBD, Drupal como sistema gestor de contenido en su versión 7 pues la propuesta de solución se integrará con sitios web desarrollados con este CMS, Visual Paradigm 8.0 como herramienta CASE para el modelado de los artefactos del análisis y las herramientas Apache JMeter y Acunetix Web Vulnerability Scanner para comprobar el rendimiento general y el cumplimiento de las especificaciones del sistema al ser sometidos a diferentes pruebas.

Capítulo II. Análisis y diseño del módulo para la publicación de contenidos web vía correo electrónico.

En el capítulo se describe la propuesta de solución, se hace un levantamiento de los requisitos funcionales y no funcionales del sistema. Se realiza el análisis y diseño del módulo. Se presentan los principales artefactos generados en las primeras fases del desarrollo.

2.1. Propuesta de solución

El módulo que conforma la propuesta de solución permite publicar contenidos web vía correo electrónico en el portal de CubaSí. Para ello el usuario con los privilegios correspondientes para acceder a las funcionalidades del módulo debe realizar los siguientes pasos para su configuración:

Paso 1:

En este paso el usuario con privilegios de administración podrá crear una cuenta en el módulo con los datos necesarios para acceder a una dirección de correo electrónico, a través de la cual obtendrá la información que se publicará, para configurar esta cuenta el usuario debe:

- Acceder al enlace Publishmail encontrado en el menú de administración.
- El sistema mostrará la página principal de configuración del módulo.
- El usuario dará *clic* al enlace Cuenta el cual mostrará la página de configuración de las cuentas del módulo para acceder al servidor web.
- El usuario seleccionará la opción Adicionar que le permitirá acceder a la ventana con los campos requeridos para crear la cuenta y debe definir los siguientes campos:
 - Nombre de la cuenta.
 - Las direcciones de correos electrónicos de los usuarios autenticados en el sitio que podrán publicar contenidos vía correo.
 - La dirección de correo que recepcionará los correos de los usuarios autenticados con permiso para publicar contenido vía correo electrónico.
 - La contraseña de la dirección de correo que recepcionará la información a publicar.
 - El puerto, dominio y los comandos extras (permite definir el certificado SSL para asegurar que los datos son enviados al servidor correcto), en correspondencia con el servidor de correo.

Paso 2:

Una vez creada la cuenta del módulo Publishmail el usuario autenticado debe acceder al módulo Feed para configurar los parámetros necesarios para mapear los datos obtenidos de los correos recepcionados con los campos del tipo contenido web que se desea crear:

Para configurar el módulo Feed se debe:

- Asignarle la cuenta anteriormente creada en el módulo Publishmail y el tipo de contenido con el que se crearán los nodos.
- Definir qué datos obtenidos del correo se mapearán con los campos del contenido como, por ejemplo:
 - El asunto del correo se importará como el título de la publicación.
 - El archivo adjunto del correo se importará como el campo “archivo” del contenido web.
 - El cuerpo del correo se importará como el cuerpo del nodo.

Después de realizar estas operaciones cada vez que se ejecuta el Cron del sistema, el módulo accede al correo y comprueba las siguientes características:

- Existe en el buzón de entrada del correo electrónico algún mensaje no leído.
- Las direcciones de los mensajes recibidos deben coincidir con las autorizadas en la cuenta.
- Con la ayuda del módulo Feeds se importan los datos del correo.
- Al obtener el contenido del correo el módulo Feeds enlazará estos datos con los elementos del tipo de contenido especificados para la importación de datos desde el correo electrónico y crea los nodos o entidades en el sitio web.

Los nodos creados no se publicarán automáticamente en el sitio, pues deben cumplir con las políticas de publicación del portal web, por lo que deben ser revisados previamente a su publicación por usuarios autorizados para realizar esta operación. Si el contenido web cumple con las características requeridas podrá publicarse en el sitio.

2.2. Modelo de dominio

Un modelo de dominio captura los tipos más importantes de objetos en el contexto del sistema. Los objetos del dominio representan los eventos que suceden en el entorno en el que trabaja el sistema. En el modelo de dominio se describe mediante diagramas UML (especialmente mediante diagramas de clases (Larman, 2016)).

Se decide realizar un modelo de dominio pues el proyecto no cuenta con un entorno de negocio definido y lo que existe en la propuesta de solución son clases que se relacionan entre sí.

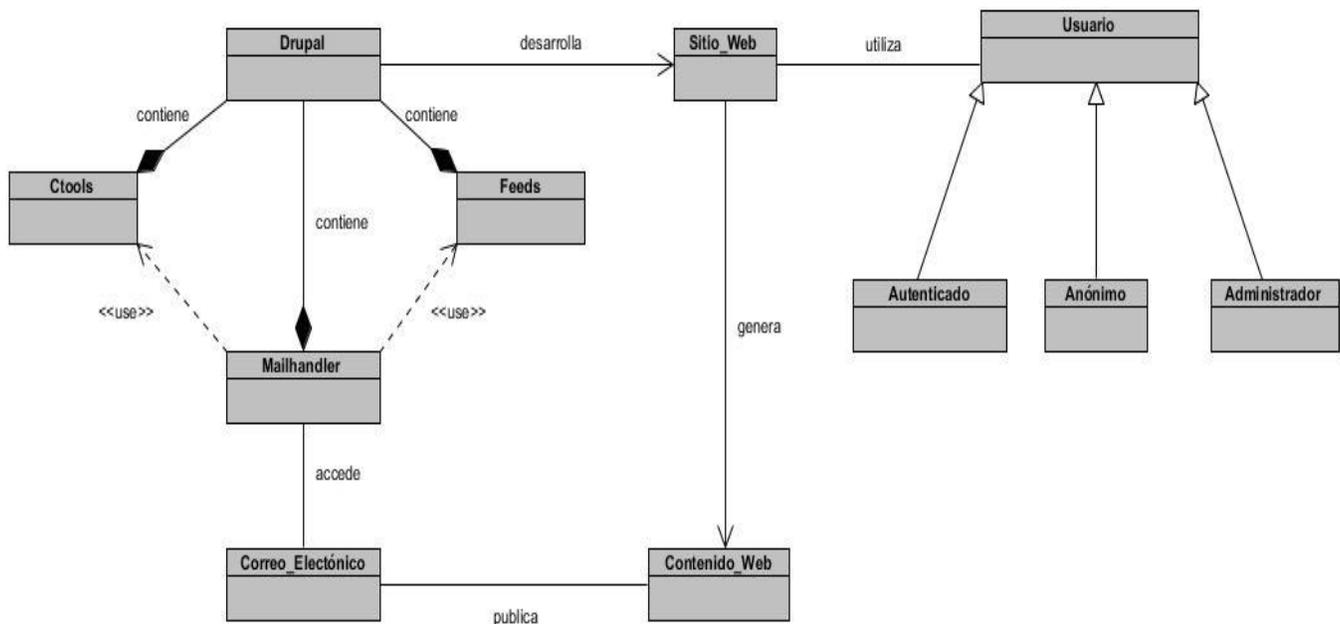


Figura 1. Modelo de dominio

2.2.1. Definición de conceptos.

- **Drupal:** Sistema gestor de contenido utilizado.
- **Sitio Web:** Sitios Web desarrollados en Drupal 7.
- **Contenido Web:** Contenidos web que serán publicados vía correo electrónico en el sitio.
- **Correo electrónico:** Correo electrónico desde el que se publicarán los contenidos web.
- **Usuario:** Usuarios con diferentes roles que están presente y utilizan los sitios web.
- **Mailhandler:** Módulo que permite publicar contenidos web vía correo electrónico.
- **Ctools:** Módulo usado para gestionar herramientas en el módulo Mailhandler.
- **Feeds:** Módulo usado para parsear los datos obtenidos del correo con el módulo Mailhandler, que se deseen publicar, con los campos de un tipo de contenido.

2.3. Requisitos Funcionales

La ingeniería de requisitos del software es un proceso de descubrimiento, refinamiento, modelado y especificación. Se refinan en detalle los requisitos del sistema y el papel asignado al software. El análisis de requisitos es una tarea de ingeniería del software que cubre el vacío entre la definición del software a nivel sistema y el diseño de software. El análisis de requerimientos permite al ingeniero de sistemas

especificar las características operacionales del software (función, datos y rendimientos), indica la interfaz del software con otros elementos del sistema y establece las restricciones que debe cumplir el software (Pressman, 2001).

A continuación, se definen un grupo de requisitos funcionales (RF) que se determinaron para el desarrollo del módulo:

Tabla 3. Requisitos funcionales

ID	Descripción	Prioridad	Complejidad
RF1	Crear cuenta	Alta	Media
RF2	Editar cuenta	Alta	Media
RF3	Eliminar cuenta	Baja	Baja
RF4	Listar cuenta	Media	Baja
RF5	Clonar cuenta	Alta	Media
RF6	Verificar datos de cuenta	Alta	Alta
RF7	Asignar permisos	Alta	Media
RF8	Editar permisos	Media	Media
RF9	Eliminar permisos	Baja	Baja
RF10	Revisar los correos electrónicos	Alta	Alta
RF11	Revisar los contenidos web	Alta	Media
RF12	Editar contenido web	Media	Media
RF13	Publicar contenido	Alta	Media
RF14	Migrar contenido	Alta	Alta
RF15	Mostrar ayuda del módulo	Media	Media

2.3.1 Historias de usuarios

Entre las técnicas ágiles que utiliza AUP se encuentra el Modelado ágil, se hará uso de esta técnica para los proyectos que necesiten por sus características encapsular sus requisitos funcionales en Historias de usuarios (Sánchez, 2015). Los tiempos de estimación y real, así como las prioridades dadas a los

requisitos establecidas en correspondencia con la importancia de la funcionalidad en cuestión se muestran en las Historias de Usuarios.

A continuación, se muestran las historias de usuario utilizadas en la propuesta de solución, el resto se muestran en los Anexos:

Tabla 4. H.U. Crear Cuenta

Número: 1	Nombre del requisito: Crear Cuenta
Programador: Bárbara B. Somoza Leal	Iteración Asignada: Iteración 1
Prioridad: Alta	Tiempo Estimado: ½ día
Riesgo en Desarrollo: 1.-Atraso en la entrega del diseño de interfaz del módulo. 2.-Falla del fluido eléctrico. 3.-Fallo en la red.	Tiempo Real: ½ día
Descripción: Los usuarios autenticados y con los permisos necesarios para crear nuevas cuentas para la publicación de contenidos web vía correo electrónico en el sistema, deben llenar los siguientes campos: Direcciones autorizadas (Campo obligatorio. Campo de texto. Longitud mínima 5 caracteres y máxima 60 caracteres. Permite los caracteres: a-z, A-Z, 0-9, y @ solo una vez. Es obligatorio que posea el carácter @ y al menos un punto en la cadena que sigue al @, pero no seguido de esta.) Dirección de correo (Campo obligatorio. Campo de texto. Longitud mínima 5 caracteres y máxima 60 caracteres. Permite los caracteres: a-z, A-Z, 0-9, y @ solo una vez. Es obligatorio que posea el carácter @ y al menos un punto en la cadena que sigue al @, pero no seguido de esta.) Contraseña (Campo obligatorio. Campo de texto. Longitud mínima 8 caracteres y máxima 60 caracteres. Permite todos los caracteres.) Protocolo (Campo obligatorio. Campo de Selección) El usuario debe seleccionar el protocolo que desee usar.	

Puerto (Campo obligatorio. Campo numérico. Longitud 3. Permite solo caracteres numéricos.) El campo se encuentra configurado por defecto con el puerto 993 para el protocolo IMAP y con el puerto 143 POP 3.

Dominio (Campo obligatorio. Campo de texto. Longitud mínima 8 caracteres y máxima 60 caracteres. Permite todos los caracteres.) Se especifica el dominio que será utilizado. Ejemplo: imap.uci.cu

Comandos extras (Campo obligatorio. Campo de texto. Longitud mínima 8 caracteres y máxima 60 caracteres. Permite todos los caracteres.) Parámetros adicionales para enviar con `imap_open ()` para determinados servidores de correo que requieren ciertos parámetros. Por ejemplo, advertencia: `imap_open () [function.imap-open]: No se pudo abrir la secuencia - esta advertencia requiere comandos Extra para tener: / novalidate-cert / tls.`

Después de llenar todos los campos el usuario puede finalizar la creación de la cuenta dando *click* en el botón Guardar, posteriormente el sistema emitirá un mensaje notificando si se ha creado satisfactoriamente la cuenta o no.

Observaciones:

1- Si el usuario introduce la información de forma correcta, el sistema emite un mensaje notificando que se ha creado satisfactoriamente la cuenta.

2-Si el usuario introduce la información de forma incorrecta, el sistema emite un mensaje notificando el error.

Prototipo de interfaz:

Direcciones autorizadas *

Los correos de los usuarios autenticados en el sitio

Dirección de correo

Introduzca la dirección de correo desde la cual se postearán los contenidos.

Contraseña

Introduzca la contraseña del correo.

Protocolo

Defina el protocolo.

Puerto

El puerto para acceder al correo puede ser 995 o 110 para POP3, 993 o 143 para IMAP).

Dominio

El dominio del servidor de correo.

Comandos extras

Defina los comandos extras, puede consultar la información en [imap_open](#).

Tabla 5. H.U. Migrar contenido

Número: 14	Nombre del requisito: Migrar contenido.
Programador: Bárbara B. Somoza Leal	Iteración Asignada: Iteración 1
Prioridad: Alta	Tiempo Estimado: ½ día
Riesgo en Desarrollo: 1.-Atraso en la entrega del diseño de interfaz del módulo.	Tiempo Real: ½ día

<p>2.-Falla del fluido eléctrico. 3.-Fallo en la red.</p>	
<p>Descripción:</p> <p>Para migrar los contenidos del correo electrónico, el usuario con permisos para administrar las funcionalidades del módulo debe realizar las siguientes operaciones:</p> <ul style="list-style-type: none"> • Configurar correctamente la cuenta del módulo para la publicación de contenidos vía correo. • Crear en el módulo Feeds un nuevo tipo de importación, al cual se le asignará la cuenta anteriormente creada y el tipo de contenido con el que se mapearán los campos del correo con los del contenido. <p>Después de realizar estas operaciones cada vez que se ejecuta el Cron del sistema, el módulo accede al correo y comprueba las siguientes características:</p> <ul style="list-style-type: none"> • Existe en el buzón de entrada del correo electrónico algún mensaje no leído. • Las direcciones de los mensajes recibidos deben coincidir con las autorizadas en la cuenta. • Con la ayuda del módulo Feeds se importan los datos del correo. <p>Al obtener el contenido del correo, el módulo Feeds parsea estos datos con los elementos del tipo de contenido especificados para la importación de datos desde el correo electrónico y crea los nodos o entidades en el sitio web.</p>	
<p>Observaciones:</p> <p>1- Si el sistema realiza satisfactoriamente la importación desde el correo se crearán los nuevos nodos. 2-Si se realiza la importación de forma incorrecta, el sistema emite un mensaje notificando el error.</p>	
<p>Prototipo de interfaz: No aplica</p>	

2.4. Requisitos no funcionales

Un requisito no funcional o atributo de calidad es en la ingeniería de software, un requisito que especifica criterios que pueden usarse para evaluar la operación de un sistema en lugar de sus comportamientos específicos, ya que éstos corresponden a los requisitos funcionales. Por tanto, se refieren a todos los

requisitos que no describen información a guardar, ni funciones a realizar, sino características de funcionamiento (Pressman, 2001).

A continuación, mostramos un grupo de requisitos no funcionales (RNF) que se determinaron para el módulo a desarrollar:

Interfaz

RNF1: El menú de configuración del módulo presentará una interfaz fácil de usar y una configuración sencilla, utilizando colores claros, en tonalidad azul.

RNF2: Los textos serán presentados en tipografías Times new roman, Arial y Verdana, además se resaltarán los encabezados en negrita.

Usabilidad

RNF3: Se debe lograr una estructura simple y lo más intuitiva posible para el bloque de suscripción.

RNF4: Se utilizarán tamaños de letra mínimo de 16px para personas con discapacidad visual.

Software

RNF5: El sistema funcionará en un entorno de trabajo compuesto por: Navegadores web Chrome 48, Firefox 44, Opera 34 e Internet Explorer a partir de su versión 10.

Seguridad

RNF6: El sistema será utilizado solo por los usuarios con permisos para administrar las funcionalidades del módulo y por el usuario administrador.

RNF7: Los errores deben mostrar la menor cantidad de código y de variables posibles, para evitar que la información comprometa la seguridad e integridad del sistema.

RNF8: Garantizar que las funcionalidades del módulo se muestren de acuerdo al rol del usuario que este activo.

Rendimiento

RNF9: El sistema debe responder en un máximo de 5 segundos las solicitudes de los usuarios

RNF10: El sistema debe soportar un mínimo de 150 peticiones concurrentes.

Hardware

RNF11: El servidor de aplicaciones web y el de base de datos, deben poseer como mínimo un CPU Dual Core 2.20.

2.5. Arquitectura de software

En el desarrollo de la propuesta de solución se utiliza la arquitectura basada en componentes pues se enfoca en la descomposición del diseño en componentes funcionales o lógicos que expongan interfaces de comunicación bien definidas, esto provee un nivel de abstracción mayor que los principios de orientación por objetos y no se enfoca en asuntos específicos de los objetos como los protocolos de comunicación y la forma como se comparte el estado (Software, 2017).

El estilo de arquitectura basado en componentes tiene las siguientes características (Agusangeli, 2015):



Figura 2. Características de la arquitectura basada en componentes

Estas características permiten que el módulo para la publicación de contenidos web vía correo pueda realizar las principales funcionalidades que operarán sobre los nodos (tipos de contenidos) para crear los contenidos web vía correo y también permite el enlace con los módulos que le otorgan el correcto funcionamiento a la propuesta de solución, como por ejemplo Feeds y Ctools.

El objetivo del uso de esta arquitectura de software en el desarrollo de la propuesta de solución es construir una aplicación compleja mediante ensamblado de módulos (componentes) que han sido previamente diseñados por otros desarrolladores a fin de ser reusados en múltiples aplicaciones, esto se puede presenciar en la utilización de las características de la propuesta de solución para asignar los permisos de usuarios a través del gancho de privilegios (hook permission), también se utilizan los ganchos de menús (hook menu), que reciben y gestionan todas las peticiones efectuadas para publicar contenidos web vía correo y por el bloque de administración para el módulo que realiza esta funcionalidad.

2.6. Patrones de diseño

Los patrones de diseño son los principios generales de soluciones que aplican ciertos estilos que ayudan a la creación de software. Para que una solución sea considerada un patrón debe comprobarse su efectividad resolviendo problemas similares en ocasiones anteriores. Otra característica es que la solución

debe ser reutilizable, lo que significa que es aplicable a diferentes problemas de diseño en distintas circunstancias (Pressman, 2007).

2.6.1. Patrones de diseño GoF

Los patrones GoF, describen las formas comunes en que diferentes tipos de objetos pueden ser organizados para trabajar unos con otros. Tratan la relación entre clases, la combinación de clases y la formación de estructuras de mayor complejidad. Permiten crear grupos de objetos para ayudar a realizar tareas complejas (Márquez, 2013).

Drupal 7 hace uso de varios patrones de diseño, estos permiten diseñar sistemas seguros y que a su vez cumplan con los estándares de diseño establecidos por normas internacionales para el desarrollo de aplicaciones web según (Mifsud, 2011). Entre los que podemos mencionar:

- **Bridge (Puente):** El módulo de la propuesta de solución fue implementado independiente al sistema de base de datos lo que permitió que algunos *hooks* pudieran realizar consultas a la base de datos sin la necesidad de hacer cambios en el código del módulo, entre los que se encuentran el *hook_menu* para definir el *array* de elementos que serán añadidos al menú de la cuenta, *hook_permission* para que otros usuarios puedan acceder a las funcionalidades del módulo, *hook_node_insert* para verificar cuando un nuevo nodo es agregado y *hook_node_update* actualizar los nodos.
- **Comando (Command):** En el desarrollo de la solución propuesta fue empleado el patrón Command a fin de que los módulos implementados no tengan que definir cada *hook*, sino sólo los que vayan a aplicar. De hecho, el propio sistema de *hook* utiliza este patrón, por lo que los módulos no tienen que definir cada *hook*, sino más bien sólo los que se necesiten implementar.
- **Cadena de responsabilidad (Chain of Responsibility):** En el desarrollo de la propuesta de solución se utilizarán módulos como por ejemplo *CTool* y *Feeds* que tendrán la tarea de gestionar solicitudes, por lo que el menú del sistema determina si hay un módulo para gestionar la solicitud, si el usuario tiene acceso a los recursos solicitados y qué función se llama para hacer el trabajo, siguiendo así el flujo de trabajo de este patrón hasta que un módulo se encarga de la petición, un módulo niega el acceso para el usuario o la cadena se ha agotado. La implementación del *hook_menu* es la evidencia de la utilización de este patrón en la propuesta de solución.

2.6.2. Patrones de diseño GRASP

Los patrones de diseño GRASP (General Responsibility Assignment Software Patterns) son descripciones formales de principios fundamentales del diseño orientado a objetos y de la asignación de responsabilidades, representan los aspectos fundamentales del diseño (Goñi, et al., 2013).

En la propuesta de solución se utilizan los siguientes patrones GRASP:

Controlador: Es un objeto (clase) donde se ejecuta un evento o todas las operaciones del sistema, una vez que el controlador recibe un evento, puede delegar sobre otros objetos para no verse sobrecargado. Un ejemplo en el que es utilizado este patrón en la propuesta de solución es en la clase *publishmailDefault.module* que se encarga de validar las direcciones de correo electrónico que podrán publicar contenido web vía correo.

Alta cohesión: Permite asignar una responsabilidad de manera que la cohesión permanezca alta, este patrón se evidencia en la clase *PublishmailPhplmapRetrieve.class.php*, pues es la encargada de acceder a los correos electrónicos y obtener la información referente a ellos, para poder ser parseados con los campos de un tipo de contenido.

Bajo acoplamiento: El acoplamiento mide el grado en que una clase está conectada a otra, tiene conocimiento de otra o, de alguna manera, depende de otra. Este patrón se puede presenciar en la clase *publishmail_default.feeds_importer_default.inc*, pues es la encargada de realizar el parseo de los datos obtenidos por la clase *PublishmailPhplmapRetrieve.class.php*, con los campos de un tipo de contenido.

2.7. Modelo de despliegue

El diagrama de despliegue, representa de forma visual las relaciones físicas que existen entre los componentes de software y hardware en el sistema (Apache, 2016).

El cliente mediante un navegador web podrá conectarse a través del protocolo HTTPS y el puerto 443 al servidor web Apache, donde se encuentra hospedado el módulo que conforma la propuesta de solución.

Una vez accedido al servidor web se podrá conectar a los siguientes servidores:

- **Servidor de base de datos:** A través de la interfaz de base de datos el servidor web se podrá acceder a este servidor mediante el protocolo TCP y el puerto 3306. Este servidor contiene al SGBD MySQL.

- **Servidor de correo:** Mediante la librería php_imap perteneciente al servidor web, se podrá acceder al servidor de correo IMAP/POP 3, a través del protocolo SMTP y el puerto 465.

A continuación, se muestra el diagrama de despliegue propuesto para el sistema, donde quedan definidas las relaciones entre los diferentes nodos físicos que lo componen.

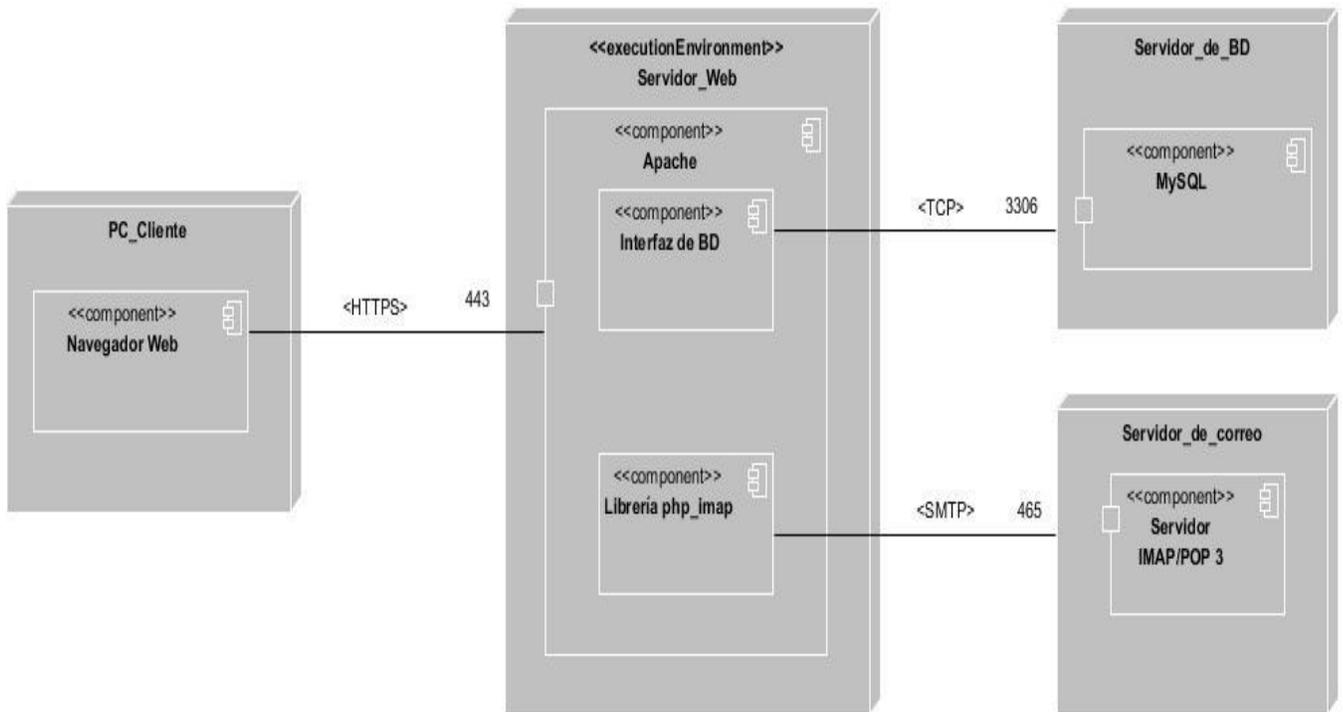


Figura 3. Modelo de despliegue

2.7.1. Descripción de los artefactos:

Para una mejor comprensión del diagrama mostrado, se describen a continuación cada uno de los artefactos que intervienen en el mismo:

PC_Cliente: Se comunica con el servidor web donde se encuentra funcionando el sitio web, mediante el protocolo HTTPS.

Navegador Web: Es un software, aplicación o programa que permitirá el acceso al sitio web que contendrá la propuesta de solución y al servidor web a través del protocolo HTTPS y el puerto 443.

Servidor_Web: En este servidor se hospeda el funcionamiento del sitio web que contendrá la propuesta de solución. El servidor de aplicaciones web es el que permite que el cliente interactúe y tenga acceso al módulo.

Interfaz de BD: Permitirá establecer la conexión con el servidor de BD, a través del protocolo TCP y el puerto 3306.

Servidor_de_BD: En él se guardan los datos correspondientes a las solicitudes realizadas por el usuario.

MySQL: Es el SGBD utilizado por el sitio web.

Librería php_imap: Permitirá establecer la conexión con el servidor de correo, a través del protocolo SMTP y el puerto 465.

Servidor de correo: Este componente contendrá el servidor de correo IMAP/POP 3, desde el cual se importarán los contenidos que serán publicados en el sitio.

Servidor de correo IMAP/POP 3: Este servidor contendrá los correos electrónicos que se importarán como contenidos.

2.7.2. Descripción de los protocolos

A continuación, se describen los protocolos utilizados para establecer las conexiones con los servidores:

<TCP> Mediante este protocolo se establece la conexión entre el servidor web y el servidor de base de datos. Este permite realizar las peticiones y obtener respuestas a las mismas de forma rápida.

<SMTP> Mediante este protocolo se establece la conexión entre el servidor web y el servidor de correo. Este permite autenticarse en el sistema mediante cuentas de usuario, así como el envío de notificaciones.

<HTTPS> Mediante este protocolo se logra establecer una conexión segura entre la PC cliente y el servidor web.

2.8. Conclusiones parciales

En el capítulo se definieron las características fundamentales de la propuesta de solución, además se describieron los requisitos funcionales y no funcionales que debe cumplir el sistema para satisfacer las necesidades del cliente, entre otros aspectos por lo que se arriba a las siguientes conclusiones:

- La descripción detallada de los requisitos funcionales y no funcionales obtenidos a partir de las técnicas de recopilación garantizó que la solución respondiera a las necesidades del cliente, sirviendo de guía para el desarrollo de las distintas funcionalidades del módulo y permitió una mejor comprensión de los resultados que se esperan alcanzar y sirvieron de guía para la codificación del sistema.
- La selección de la arquitectura, los artefactos generados y los patrones de diseño utilizados por Drupal constituyeron una guía fundamental para el desarrollo de la propuesta de solución a través de los estándares web.
- Los diagramas propuestos en este capítulo constituyeron una guía primordial para dar una mayor descripción de los elementos del sistema y la construcción del mismo. También permitieron ampliar las funcionalidades del módulo y disminuir el impacto de los cambios futuros en el mismo.

Capítulo III. Implementación y validación del Módulo para la publicación de contenidos web vía correo electrónico

3.1. Introducción

En el capítulo se representa el diagrama de componentes, se describen los estándares de codificación utilizados durante la implementación. También se realizan los casos de pruebas pertinentes para darle validez a los requerimientos funcionales y garantizar el óptimo funcionamiento de la aplicación. Además, se utilizan herramientas de seguridad y rendimiento para ejecutar pruebas de este tipo al sistema y se realiza una valoración de los resultados obtenidos al aplicar las pruebas.

3.2. Diagrama de componentes

El diagrama de componentes describe la descomposición física del sistema de software (y, eventualmente, de su entorno organizativo) en componentes (Campderrich, 2002). En la figura 4 se presenta el diagrama de componente del módulo para la publicación de contenido web vía correo electrónico en el portal de CubaSí.

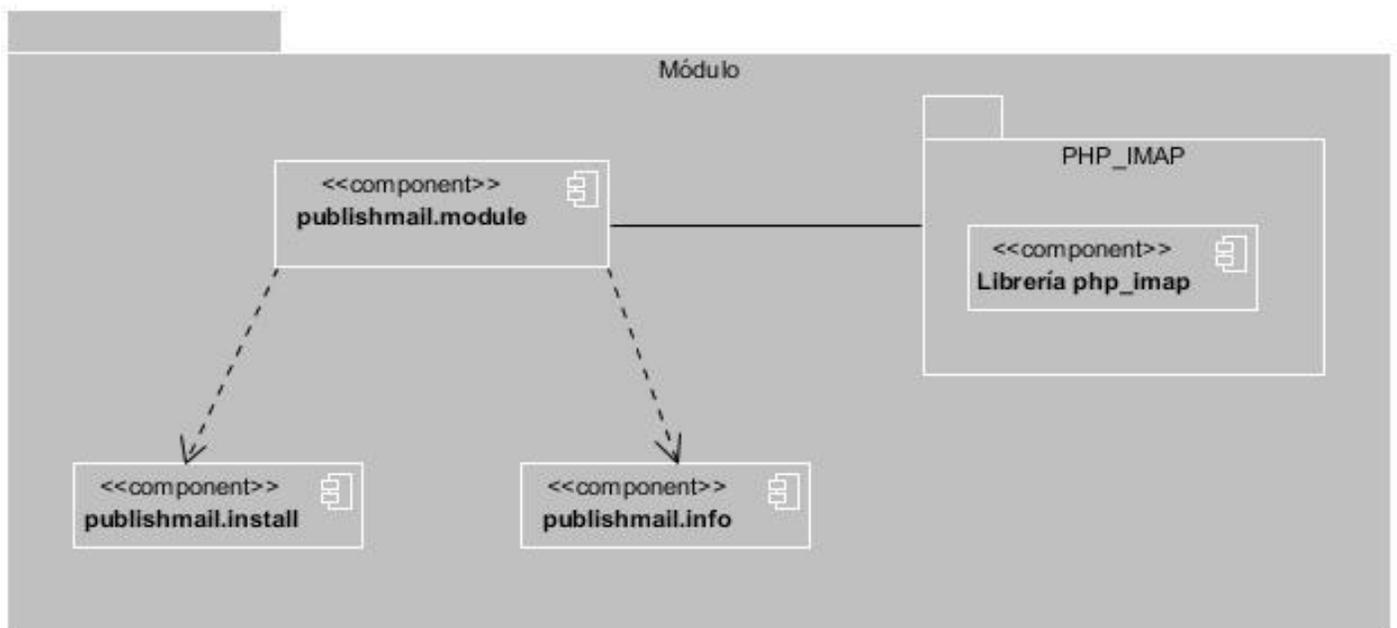


Figura 4. Diagrama de componentes

3.2.1. Descripción de los componentes

A continuación, se describen los componentes del diagrama:

publishmail.module: Archivo que contiene las llamadas a las funciones hooks de Drupal.

publishmail.info: Contiene la información básica del módulo.

publishmail.install: Archivo donde se implementan las instancias del módulo.

Librería php_imap: Permitirá establecer la conexión con el servidor de correo para obtener los datos de los correos electrónicos.

3.3. 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. Para el desarrollo del módulo se utilizan los estándares de codificación definidos por Drupal en (Drupal.org, 2017). A continuación, se explican y ejemplifican los estándares de codificación utilizados en la propuesta de solución.

Indentación y espacios en blanco

Consiste en insertar espacios en blanco o tabuladores en determinadas líneas de código para facilitar su comprensión. Se usa una indentación de dos espacios en blanco sin usar tabuladores, se evita usar espacios en blanco al final de cada línea.

```
22 function publishmail_permission() {
23     return array(
24         'administer publishmail' => array(
25             'title' => t('Administer Publishmail'),
26         ),
27     );
28 }
29
```

Figura 5. Ejemplo de indentación y espacios en blanco

Operadores

Todos los operadores binarios como: +, -, !, =, ==, >, deben tener un espacio antes y después del operador, los operadores unarios como ++, no utilizan espacios entre el operador y la variable.

```

19 function publishmail_mailregister() {
20     $variables = array(
21         'message' => 'Bienvenido al modulo Publishmail',
22     );
23     return theme('publishmail_mailregister', $variables);
24 }
25

```

Figura 6. Ejemplo de operadores

Estructuras de control

Las estructuras de control incluyen if, for, while, swwitch, entre otros. Las mismas deben presentar un espacio entre la palabra de control (if, for) y el paréntesis que abre, para distinguirlos de las llamadas a las funciones. La llave de apertura ({} se situará en la misma línea que la definición de la estructura de control, separada por un espacio como se demuestra en la figura 7.

```

183 function publishmail_plugin_load_class($module, $plugin, $type, $id, $args = NULL) {
184     ctools_include('plugins');
185     if ($class = ctools_plugin_load_class($module, $type, $plugin, 'handler')) {
186         return new $class($args);
187     }
188 }

```

Figura 7. Ejemplo de estructuras de control

Además, se debe usar siempre las llaves {} aún en los casos en que no sea obligatorio su uso y las estructuras else y elseif se escribirán en la línea siguiente al cierre de la sentencia anterior como se demuestra en la siguiente figura.

```

127     if ($field) {
128         $field['cardinality'] = -1; // Unlimited
129         field_update_field($field);
130     }
131     else {
132         migrate_postMail_mails_image();
133     }
134 }

```

Figura 8. Ejemplo de la estructura if y else

La estructura switch se utilizará de la siguiente manera:

```
51 function publishmail_block_view($delta = '') {
52     switch ($delta) {
53         case 'publishmail':
54             $block['content'] = publishmail_mailregister();
55             return $block;
56     }
57 }
```

Figura 9. Ejemplo de la estructura switch

Longitud de líneas

En general las líneas de código no deben contener más de 80 caracteres.

```
224 function publishmail_mailregister_load($mail) {
225     ctools_include('export');
226     return ctools_export_crud_load('publishmail_mailregister', $mail);
227 }
228
```

Figura 10. Ejemplo de longitud de línea menor a 80 caracteres

Las líneas que contienen nombres de funciones extensos, definiciones de clases o funciones, declaraciones de variable, pueden exceder los 80 caracteres como se demuestra a continuación.

```
47 function migrate_postMail_mails_uninstall() {
48     if ($vids = taxonomy_vocabulary_load_multiple(array(), array('machine_name' => 'migrate_postMail_mails_styles'))) {
49         // Grab key of the first returned vocabulary.
50         taxonomy_vocabulary_delete(key($vids));
51     }
52     migrate_postMail_mails_content_type_delete();
53 }
```

Figura 11. Ejemplo de longitud de línea mayor a 80 caracteres

Llamadas a funciones

Las funciones deben ser llamadas sin espacios entre el nombre de la función, el paréntesis que abre y el primer argumento de la función. Se separan los argumentos por un espacio, y no se utilizan espacios entre el último parámetro, el paréntesis que cierra y el punto y coma.

```
23 return theme('publishmail_mailregister', $variables);
```

Figura 12. Ejemplo de llamadas a funciones

Arreglos (arrays)

Los valores dentro de un array (o matriz) se deben separar por un espacio (después de la coma que los separa). El operador => debe separarse por un espacio a ambos lados. Cuando la línea de declaración del array supera los 80 caracteres, cada elemento se debe escribir en una única línea, indentándolo una vez (2 espacios). En este último caso, la coma de separación del último elemento también se escribirá, aunque no existan más elementos como se demuestra en la siguiente figura.

```
29 function publishmail_theme() {
30     return array(
31         'publishmail_mailregister' => array(
32             'template' => 'publishmail_mailregister',
33         ),
34     );
35 }
```

Figura 13. Ejemplo de array

Uso de comillas

Se pueden usar tanto las comillas simples ('cadena') como las comillas dobles ("cadena") para delimitar las cadenas de caracteres. Las comillas dobles son necesarias si se desean incluir variables dentro de las cadenas de texto o cuando el texto puede incluir alguna comilla simple.

```
171 $aids = trigger_get_assigned_actions('publishmail_auth_failed');
```

Figura 14. Ejemplo de uso de comillas simples

Etiquetas de apertura y cierre de PHP

Se omite la etiqueta de cierre de PHP: (?>) al final de los archivos: module, inc, install.

```

1 <?php
2
3 /**
4  * @file
5  * Set up the migration postMail module.
6  */
7
8 require_once DRUPAL_ROOT . '/' . drupal_get_path('module', 'migrate_postMail') .
9   '/mails.install.inc';
10

```

Figura 15. Ejemplo de apertura y cierre de un archivo .inc

Utilización del punto y coma (;) en código PHP

En Drupal es obligatorio el terminador de línea (;).

```

89 $help = file_get_contents(dirname(__FILE__) . "/INSTALL.txt");
90 return _filter_autop($help);

```

Figura 16. Ejemplo del terminador (;)

Nombres de funciones

Los nombres de las funciones deben estar escritos en minúsculas y las palabras separadas por un guión bajo. Se debe incluir siempre como prefijo el nombre del módulo. En su declaración, después del nombre de la función, el paréntesis de inicio de los argumentos debe ir sin espacio. Cada argumento debe ir separado por un espacio, después de la coma del argumento anterior.

```

86 function publishmail_help($path, $arg) {

```

Figura 17. Ejemplo de nombre de funciones

Nombres de variables

Las variables deben ser nombradas en minúsculas y las palabras deben estar separadas por un guión bajo.

```

54 $block['content'] = publishmail_mailregister();

```

Figura 18. Ejemplo de nombre de variables

Variables persistentes

Las variables persistentes definidas por Drupal al usar las funciones, `variable_get ()` / `variable_set ()` deben ser nombradas en minúsculas y las palabras deben estar separadas por un guión bajo, como prefijo deben tener el nombre del módulo para evitar colisiones.

```
261 drupal_set_message(t($message, $variables), $type);
```

Figura 19. Ejemplo de variable persistente

Constantes

Los nombres de las constantes deben escribirse en mayúsculas, con guiones bajos para separar palabras.

```
61 define('PUBLISHMAIL_MENU_PREFIX', 'admin/structure');
```

Figura 20. Ejemplo de constantes

3.4. Validación de la propuesta de solución

Las pruebas son el conjunto de técnicas que permiten determinar la calidad de un producto de software. Se integran dentro de las diferentes fases del ciclo de vida del software dentro de la ingeniería de software. De esta manera, se ejecuta un programa con el objetivo de descubrir los errores que presenta. Se deben efectuar unas medidas o pruebas que permitan comprobar el grado de cumplimiento respecto de las especificaciones iniciales del sistema. Una prueba tiene éxito si descubre un error no detectado hasta entonces (Cibertec, 2014).

Se considera que las evaluaciones de Satisfactoria e Insatisfactoria de los indicadores de rendimiento, integración y seguridad estarán en correspondencia con los resultados de las pruebas de integración, seguridad y rendimiento realizadas al módulo.

3.4.1. Pruebas funcionales

Las pruebas funcionales se basan en los requerimientos entregados para el desarrollo de un software. Estas se realizan mediante casos de prueba cuyo fin es validar que el software cumple con el nivel de calidad requerido para entrar en producción (León, 2015).

Para la ejecución de este tipo de pruebas, suelen emplearse dos métodos fundamentales: el método de Caja Blanca y el método de Caja Negra. El primero se centra en las pruebas al código de las aplicaciones;

mientras que el segundo permite a los probadores enfocar su atención en el funcionamiento de la interfaz, a través del análisis de los datos de entrada y los de salida (Jimenez, y otros, 2014).

Los casos de prueba de caja negra pretenden demostrar que (Pressman, 2001):

- Las funciones del software son operativas.
- La entrada se acepta de forma correcta.
- Se produce una salida correcta.
- La integridad de la información externa se mantiene.

Como resultado de las pruebas de aceptación se obtendrán artefactos descritos en tablas, estas contarán con los siguientes campos:

Código: identificador de la prueba realizada sugerente a la Historia de Usuario a la que hace referencia.

Nombre: nombre de la prueba a realizar.

Nombre del probador: nombre de la persona que realiza la prueba.

Descripción: se describe la funcionalidad que se desea probar.

Condiciones de Ejecución: mostrará las condiciones que deben cumplirse para poder llevar a cabo el caso de prueba, estas condiciones deben ser satisfechas antes de la ejecución del caso de prueba para que se puedan obtener los resultados esperados.

Entradas/Pasos de Ejecución: descripción de cada uno de los pasos seguidos durante el desarrollo de la prueba, se tiene en cuenta cada una de las entradas que hace el usuario con el objetivo de ver si se obtiene el resultado esperado.

Resultado esperado: breve descripción del resultado que se espera obtener con la prueba realizada.

Evaluación de la prueba: acorde al resultado de la prueba realizada se emitirá una evaluación sobre la misma. Esta evaluación tendrá uno de los dos valores que a continuación se describen:

- Satisfactorio
- Insatisfactorio

A continuación, se muestra una propuesta de diseño de caso de prueba por historias de usuarios, las demás se pueden encontrar en los Anexos2.

Tabla 6. Caso de prueba Crear Cuenta

Caso de Prueba 1		
Código: CDPHU1	Nombre: Crear cuenta.	
Nombre del probador: Bárbara Beatriz Somoza Leal.		
Descripción: Prueba a la funcionalidad crear cuenta.		
Condiciones de Ejecución: El usuario debe estar autenticado y tener los permisos necesarios.		
<p>Entrada / Pasos de ejecución:</p> <ul style="list-style-type: none"> • El usuario accede al sistema. • El usuario da <i>clic</i> en el enlace Publishmail encontrado en el menú de administración. • El sistema mostrará la página principal de configuración del módulo. • El usuario dará <i>clic</i> al enlace Cuenta el cual mostrará la página de configuración de las cuentas del módulo. • El usuario seleccionará la opción Adicionar que le permitirá acceder a la ventana con los campos requeridos para crear la cuenta. • Después de llenar los campos requeridos el usuario dará clic en el botón Salvar encontrado al final de la ventana. • El sistema mostrará un mensaje confirmando que se ha creado correctamente la cuenta. <p>La entrada consta de la introducción de los datos en los siguientes campos:</p> <p>Correos autenticados: usuario@ejemplo.com</p> <p>Usuario: nombre_usuario</p> <p>Dirección de correo: usuario@ejemplo.com</p> <p>Contraseña: 123admin.</p> <p>Protocolo: IMAP</p> <p>Dominio: imap.ejemplo.com</p> <p>Puerto: 143</p> <p>Comandos extras: /novalidate-cert</p>		
Escenarios:	Resultados Esperados	Evaluación de la

		Prueba:
EC1. El usuario introduce correctamente los datos.	El sistema crea la cuenta y muestra el siguiente mensaje de confirmación: ``Se ha creado satisfactoriamente la cuenta``.	Satisfactoria.
EC2. El usuario introduce datos usados en otra cuenta.	El sistema no crea la cuenta y muestra el siguiente mensaje de error: ``Ha introducido datos que ya están en uso, verifique los campos de la cuenta``.	Satisfactoria.
EC3. El usuario deja uno o varios campos vacíos.	El sistema no crea la cuenta y muestra el siguiente mensaje de error: ``Ha dejado campos vacíos, verifique los campos de la cuenta``.	Satisfactoria.
EC4. El usuario introduce campos incorrectos.	El sistema no crea la cuenta y muestra el siguiente mensaje de error: ``Ha introducido campos incorrectos, verifique los campos de la cuenta``.	Satisfactoria.

Resultados de las pruebas funcionales

Para la validación de los requisitos funcionales del módulo para la publicación de contenido web vía correo electrónico se realizaron 3 iteraciones de pruebas funcionales al mismo, en las cuales se detectaron un total de 22 no conformidades, 18 en la primera iteración las cuales fueron resueltas no quedando ninguna pendiente, 4 en la segunda iteración, las cuales fueron resueltas en su totalidad. Se realizó una tercera iteración donde no se detectó ninguna no conformidad, arrojando como resultado final que el sistema funciona satisfactoriamente. En la figura se muestran los resultados obtenidos en cada una de las iteraciones de pruebas realizadas al módulo.

No Conformidades

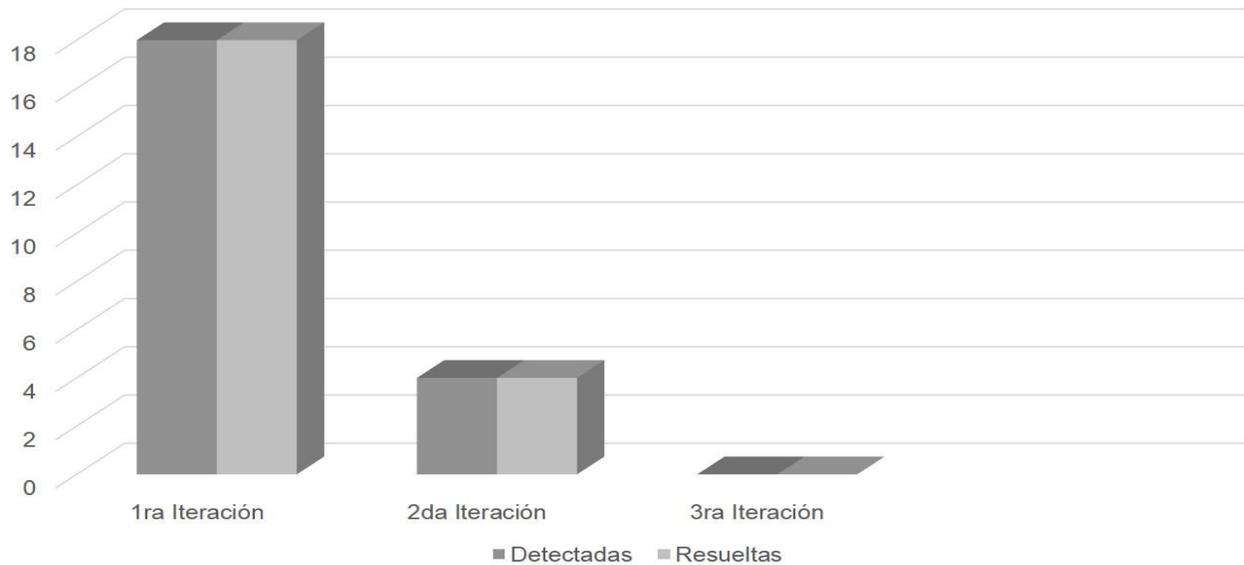


Figura 21. Comportamiento de las no conformidades por iteraciones

Algunas de las no conformidades encontradas durante el proceso de prueba fueron las siguientes:

- Datos incorrectos guardados en las bases de datos sin validación previa.
- Mensajes de error que no corresponden con los errores definidos en los casos de pruebas.
- Errores ortográficos.
- Errores de concordancia.
- Errores de idioma.
- Errores de validación de los campos.

3.4.2. Pruebas de integración

Las pruebas de integración son definidas para verificar el correcto ensamblaje entre los distintos módulos que conforman un sistema informático. Las mismas validan que estos componentes realmente funcionan juntos, son llamados correctamente y, además, transfieren los datos correctos en el tiempo preciso y por las vías de comunicación establecidas (Sommerville, 2005).

Se realizan pruebas de integración al módulo para verificar la compatibilidad y el funcionamiento de las interfaces que comunican los componentes de la propuesta de solución con el portal de CubaSí. Se utilizó

el módulo Testing (que está incluido en Drupal core) para detectar errores en el funcionamiento del sitio y del módulo cuando este realiza la importación de contenido desde un correo electrónico.

Resultados de las pruebas de integración

Atendiendo a lo anteriormente planteado en la propuesta de solución se integra el módulo para la publicación de contenido web vía correo electrónico con el portal de CubaSí desarrollado con el uso del CMS Drupal en su versión 7 y que utiliza como SGBD MySQL en su versión 5. Con la ayuda del módulo Testing se realizan *test* al funcionamiento del módulo y del sitio con los cuales se comprueba que el módulo no afecta el funcionamiento del portal web y que permite que los usuarios autenticados y con los permisos necesarios pueden acceder a todas las funcionalidades que le permitan publicar contenido web vía correo electrónico sin afectar el correcto funcionamiento del portal.

Por lo que se puede concluir que el módulo que conforma la propuesta de solución se integra satisfactoriamente con el portal de CubaSí.

3.4.3. Pruebas de seguridad

Las pruebas de seguridad garantizan que los usuarios estén restringidos a funciones específicas o que su acceso esté limitado únicamente a los datos que están autorizados a acceder. Sólo aquellos usuarios autorizados a acceder al sistema son capaces de ejecutar las funcionalidades disponibles. El objetivo fundamental de este tipo de pruebas es comprobar los niveles de seguridad lógicos del sistema (García, 2014).

Resultados de las pruebas de seguridad

Para evaluar la seguridad de la propuesta de solución se desarrollan pruebas utilizando la herramienta Acunetix Web Vulnerability Scanner. A continuación, en la tabla 7 se muestra el escaneo de vulnerabilidades para pruebas de seguridad aplicadas a la propuesta de solución.

Tabla 7. Resultado del escaneo de vulnerabilidades para pruebas de seguridad.

Categoría de vulnerabilidades	Cantidad
Formularios HTML sin protección contra ataques CSRF	23
Campos de contraseña con auto completamiento activado	0
El método TRACE está activado	1
Posibles archivos sensibles	5
Posibles directorios sensibles	2
Total de vulnerabilidades	31

Para corregir las deficiencias encontradas se realizó un profundo análisis de las mismas en el cual se determinó que las vulnerabilidades detectadas dependen del servidor web donde está alojado el módulo y al sistema Drupal, por lo que no corresponden a problemas críticos de seguridad del mismo. Se llevaron a cabo un conjunto de acciones para reforzar la seguridad en el módulo para la publicación de contenidos web vía correo electrónico, las cuales fueron:

- Para prevenir los ataques de falsificación de petición en sitios cruzados (por sus siglas en inglés, CSRF) se incluyeron campos con identificadores únicos en los formularios HTML del sistema.
- Para desactivar el método TRACE se añadió al final del fichero de configuración de apache la siguiente línea: TraceEnable off.
- Para impedir el acceso de usuarios no deseados a archivos con contenido sensible se añadió al fichero de configuración de apache la configuración que se muestra a continuación en la figura 22:

```
<FilesMatch "(\\.bak|config|dist|fla|inc|ini|log|psd|sh|sql|swp)|~)$">
    Order allow,deny
    Deny from all
    Satisfy All
</FilesMatch>
```

Figura 22. Configuración para proteger archivos sensibles

- Para prevenir los listados de directorios se removió la palabra clave Indexes de cada directiva Options en el fichero de configuración de apache, utilizando la siguiente línea: Options –Indexes.

3.4.4 Pruebas de rendimiento

Mediante las pruebas de rendimiento es posible hallar tendencias y comportamientos para los elementos de una aplicación, los cuales generan bajo rendimiento. Este tipo de pruebas permiten identificar cuellos de botella, capacidad de concurrencia de usuarios, tiempos de respuesta de operaciones de negocio a nivel de sistema, establecer un marco de referencia para pruebas futuras, determinar el cumplimiento de los objetivos de rendimiento y requerimientos no funcionales, entre otros (Torets, 2016).

Pruebas de estrés: Mediante las pruebas de estrés es posible identificar la capacidad de respuesta de un sistema bajo condiciones de carga extrema, representadas por una alta concurrencia de usuarios o procesos (Torets, 2016).

Prueba 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, determinando el tiempo de respuesta de todas las transacciones críticas del sistema (Torets, 2016).

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: Windows 8.0
- Microprocesador: Intel(R) Core (TM) i5 CPU M 480 @2.67 GHz (4 CPUs)
- Memoria RAM: 4 GB
- Disco Duro: 500 GB
- Tipo de Conexión: Ethernet 100Mbps.

Resultados y análisis de las pruebas de carga y estrés

Para realizar la validación del sistema se definen una prueba de 25 usuarios que realizan 150 solicitudes simultáneas, los cuales se conectan concurrentemente con un período entre cada solicitud de un segundo y con un acceso aleatorio a una lista de enlaces. A partir de ahí, se recolectaron los datos necesarios para la interpretación de los resultados de estas pruebas, los cuales se pueden observar en la siguiente figura.

Etiqueta	# Muestras	Media	Mediana	Linea de 90%	Mín	Máx	% Error	Rendimiento	Kb/sec
606 /predefe...	25	6092	6091	9798	2215	9846	0,00%	2,3/sec	10,8
641 /predefe...	25	5080	4866	6274	4150	6492	0,00%	1,7/sec	7,3
676 /predefe...	25	4008	4106	4614	2694	5423	0,00%	1,8/sec	7,4
779 /predefe...	25	1455	1543	1923	770	2035	100,00%	2,2/sec	21,3
813 /predefe...	25	1331	1163	1957	676	2062	100,00%	2,4/sec	23,7
914 /predefe...	25	1295	1407	1988	363	2214	100,00%	2,9/sec	28,9
Total	150	3210	2214	6274	363	9846	50,00%	6,6/sec	47,1

Figura 23. Valores obtenidos para una muestra de 25 usuarios

Para el análisis, dado que se considera que hay un gran número de muestras a analizar, no se requiere hacer la suposición de que la muestra tiene una distribución normal, ya que por el Teorema Central del Límite (TCL), para n grande implica que X tiene una distribución aproximadamente Normal sin importar la naturaleza de la distribución poblacional (Devore, 2010).

El tiempo total utilizado para la muestra se puede calcular con la siguiente fórmula (Díaz, 2010):

$$\text{Tiempo total} = \# \text{Muestras} * \text{Media} = 150 * 3210 = 481500 \text{ milisegundos}$$

El tiempo total requerido por cada usuario, se puede calcular de la siguiente manera (Díaz, 2010):

$$((\text{Tiempo total}/1000) /60) / \text{cantidad de usuarios} = ((481500/1000))/60/25 = 0,321 \text{ minutos}$$

Se analizaron los resultados a través de un intervalo de confianza con un nivel de confianza al 95%.

$$[TP - Z_{0.95} * S/\sqrt{n}; TP + Z_{0.95} * S/\sqrt{n},] \text{ Intervalo de confianza}$$

Donde:

- Tiempo promedio (TP) de respuesta es: 10254
- Estimador de desvío (S) es: $\sqrt{(\sum_{i=1}^n (X_i - X)^2) / n - 1} = 3042$
- Tamaño de la muestra:

- Z 0.95: 1.96

Quedando el **intervalo de confianza**:

[2,723; 3,697]

Esto significa que, puede esperarse un tiempo de respuesta promedio entre 2,723 y 3,697 segundos para una cantidad de 25 usuarios simultáneos realizando 150 solicitudes.

3.5. Validación de la hipótesis

Se considera que los indicadores y medidas de la hipótesis de la investigación planteada, se pueden evaluar en correspondencia con el porcentaje de evaluaciones realizadas por los expertos seleccionados, para validar la propuesta de solución:

Muy Adecuada > 50%

Bastante Adecuada > 30%

16% <Adecuada<29%

8% < Poco Adecuada < 15%

Inadecuada < 8%

Se utiliza el método criterio de expertos en su variante Delphi para la validación de la hipótesis de investigación a través del cumplimiento de los pasos siguientes (Sánchez, 2015):

Paso 1: Identificación de los posibles expertos.

En este paso se definen las características que deben cumplir los posibles expertos para su posterior selección.

Para identificar los posibles expertos se tuvieron en cuenta, la experiencia profesional en relación con la publicación de contenidos web y el desarrollo, uso y gestión de módulos en Drupal 7, de modo que cuenten con la capacidad de hacer recomendaciones, detectar deficiencias y valorar los aspectos y funcionalidades de la propuesta de solución.

Paso 2: Selección de los expertos.

Al analizar las características a tener en cuenta para la selección de los posibles expertos, se decide seleccionar siete con suficiente conocimiento en la investigación y desarrollo de módulos desarrollados para el CMS Drupal, todos graduados universitarios y una gran parte de ellos ha obtenido títulos de formación académica.

Tabla 8. Expertos seleccionados en la validación de la investigación

No.	Expertos	Entidad	Años de experiencia
1.	MSc. Huber Viltres Salas	CIDI	10
2.	MSc. Miguel Jaeger Rodriguez Lazo	CIDI	10
3.	MSc. Sahilyn Delgado Pimentel	CIDI	10
4.	Ing. José Gabriel Espinosa Ramirez	CIDI	5
5.	Ing. Olber Ríos Cutiño	CIDI	5
6.	Ing. Yuriesky Madrigal Vilches	CIDI	5
7.	Ing. Yuneldis Reyes Velázquez	CIDI	5

Paso 3: Realización de la consulta a los expertos y procesamiento y valoración de la información obtenida.

En este paso se sometió a la consideración de los 7 expertos seleccionados un instrumento para la validación del módulo, el mismo está compuesto de cuatro sentencias que permitirán conocer la opinión de los mismos. El instrumento empleado cuenta con las siguientes categorías evaluativas:

- Muy adecuado (MA).
- Bastante adecuado (BA).
- Adecuado (A).
- Poco adecuado (PA).
- Inadecuado (I).

Tabla 9. Sentencias aplicadas a los expertos

No.	Sentencias	Evaluación
1.	El módulo presenta una navegabilidad óptima, una interfaz intuitiva y de estructura simple.	
2.	Existe confidencialidad e integridad en el módulo.	
3.	El módulo permite publicar contenido web en el portal CubaSí vía correo electrónico.	
4.	La gestión de las cuentas del módulo permiten publicar contenido web en el portal CubaSí desde una única dirección de correo electrónico.	
5.	El módulo permite publicar archivos multimedia en los contenidos web.	
6.	El sistema accede a los correos electrónico especificados en las cuentas.	
7.	El módulo accede a los datos del correo electrónico para su posterior publicación.	

En un segundo momento se calcula el coeficiente de Kendall que permite analizar la concordancia en las valoraciones realizadas por los expertos. En este caso el coeficiente concordancia (W) será un índice de la divergencia del acuerdo efectivo entre los expertos. El coeficiente de concordancia de Kendall se obtiene de la expresión (SAMPIERI, 2006).

$W = 12S / K^2 (N^3 - N)$ **Coeficiente de Kendall**

$W = 0,00672$

donde S representa el cuadrado de las desviaciones medias (406), K el número de expertos (7) y N el número total de aspectos a evaluar (7). El valor de W (0,00205) oscila entre 0 y 1. El valor de 1 significa una concordancia de acuerdos total y el valor de 0 un desacuerdo total.

Se aplica además la Prueba de Significación de Hipótesis para comprobar el grado de significación de Kendall, planteándose la hipótesis nula y la alternativa de la siguiente forma:

H0: no existe concordancia entre los expertos

H1: existe concordancia entre los expertos

El cálculo del Chi cuadrado real:

$$X^2 = K(N-1) W \quad \text{Chi cuadrado}$$

$$X^2 = 0.0861$$

El resultado real calculado se compara con el tabulado en la tabla del percentil de la distribución Chi cuadrado. Para tener un 95% de confianza se utilizará $\alpha=0,05$. Si se cumple que X^2 calculada $< X^2(\alpha, N-1)$, entonces se infiere que existe concordancia de criterios entre los expertos al considerar válida la hipótesis alternativa H1. Como $0,0861 < 1.635$ se puede afirmar que existe concordancia entre los expertos.

Los criterios aportados por los expertos se someten a una prueba estadística no paramétrica que permite concluir qué valoración final tiene cada uno de los aspectos a evaluar. Para los datos anteriores se debe confeccionar una distribución de frecuencia a partir de los datos primarios para cada uno de los aspectos sometidos a consulta (Castro, 2014).

Tabla 10. Frecuencias asociadas a los parámetros evaluativos

Categorías evaluativas	Frecuencia Absoluta	Frecuencia Relativa
Muy Adecuado (MA)	25	0.51
Bastante Adecuado (BA)	16	0.33
Adecuado (A)	8	0.16
Poco Adecuado (PA)	0	0
Inadecuado (I)	0	0

Los resultados obtenidos de la validación pueden observarse en la figura 24.

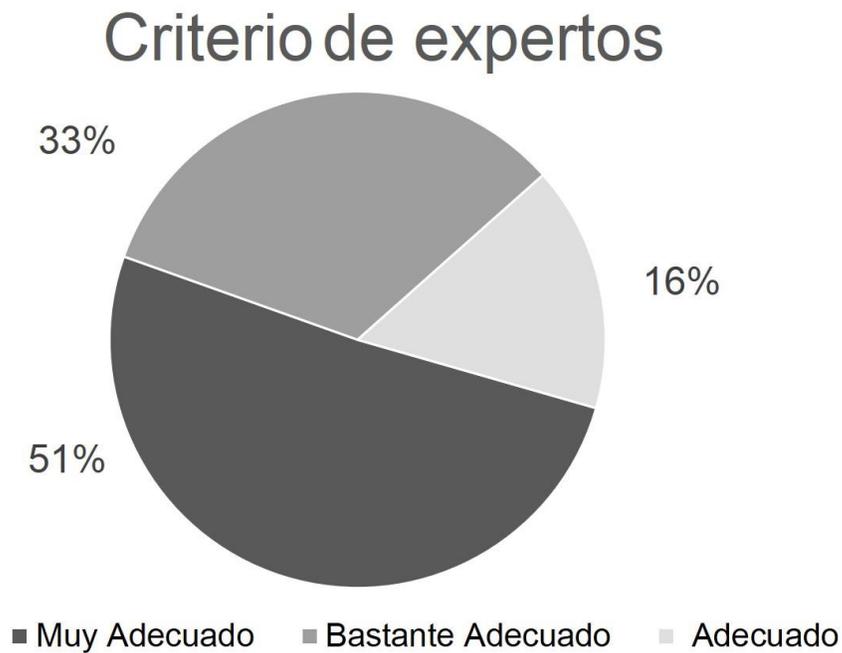


Figura 24. Comportamiento de la valoración de los expertos según los parámetros evaluativos

Al ser analizados los resultados, se puede identificar de la consulta de expertos el alto valor que tiene la utilización del módulo para la publicación de contenido web vía correo electrónico.

Se puede concluir que los indicadores expresados en la hipótesis de la investigación fueron evaluados de muy adecuado por un 51% de las evaluaciones realizadas por los expertos, un 33% de bastante adecuado y un 16% de adecuado.

3.6. Conclusiones parciales

En el capítulo se confeccionó el diagrama de componentes del sistema, se realizaron pruebas al módulo para validar la propuesta de solución y se validó la hipótesis por lo que se puede concluir lo siguiente:

- La confección del diagrama de componente permitió representar los aspectos físicos del sistema pues muestra la organización y las dependencias entre los componentes del mismo.
- La utilización de estándares de codificación de código permitió adoptar una estructura homogénea que facilita la comunicación y asegura la calidad, menos errores y un fácil mantenimiento.
- La realización de las diferentes pruebas para validar la propuesta de solución permitió detectar las deficiencias presentes, subsanarlas en el menor tiempo posible para así mejorar características como son el rendimiento, la seguridad, entro otros aspectos presentes en el módulo.
- La aplicación del método criterio de expertos utilizado para la validación de la hipótesis demostró que la propuesta de solución mejora el proceso de publicación de contenido web en el portal de CubaSí pues el módulo es considerado como muy adecuado por el 80% de los expertos consultados.

Conclusiones

En la investigación se realizó un proceso de desarrollo de software, con el objetivo de lograr mejorar el proceso de publicación de contenidos web vía correo electrónico en el portal de CubaSí. Una vez completada la investigación, se puede concluir lo siguiente:

- El estudio realizado posibilitó definir el marco teórico conceptual de la investigación, identificándose así las bases para analizar, diseñar e implementar el módulo, lo que permitió la modelación de los artefactos que contribuyeron al diseño de la propuesta de solución.
- Una vez definida la propuesta de solución e identificados los requisitos funcionales y no funcionales, el estilo arquitectónico y los patrones de diseño, el proceso de desarrollo de software garantizó que la solución respondiera a las necesidades del cliente y de los usuarios finales, sirviendo de guía para el desarrollo de las distintas funcionalidades del módulo.
- El módulo implementado permite la publicación de contenidos web vía correo electrónico en el portal de CubaSí desarrollado en el CMS Drupal en su versión 7.
- Con las diferentes pruebas y técnicas utilizadas para validar el módulo desarrollado, se detectaron y corrigieron las insuficiencias encontradas, se determinó que el módulo posee un rendimiento acorde con los resultados esperados y las pruebas de integración arrojaron que las funcionalidades del sistema no se ven afectadas con la integración del mismo al portal de CubaSí.
- La aplicación del método criterio de expertos para validar la hipótesis demostró que el módulo para la publicación de contenidos web vía correo electrónico contribuye a la mejora del proceso de publicación de contenidos web en el portal de CubaSí, pues el mismo fue evaluado de muy adecuado por los expertos seleccionados.

Recomendaciones

Se recomienda para futuras investigaciones:

- Ampliar las funcionalidades del módulo para la publicación de contenidos web vía correo electrónico como, por ejemplo, implementar una nueva funcionalidad que permita publicar videos con el uso de esta herramienta.
- Integrar el módulo a otros portales desarrollados en Drupal 7.

Bibliografía

- ABC, Definición. 2017.** Concepto en Definición ABC. *Definición de Publicación. Concepto en Definición ABC.* [En línea] 2017. [Citado el: 5 de 5 de 2017.] <https://www.definicionabc.com/general/publicacion.php>.
- Alegsa, Leandro. 2010.** Alegsa. *Módulo.* [En línea] 5 de 12 de 2010. [Citado el: 6 de 11 de 2016.] <http://www.alegsa.com.ar/Dic/modulo.php>.
- Apache, Software Foundation. 2016.** *Apache Software Foundation.* [En línea] 2016. [Citado el: 17 de 10 de 2016.] <http://jmeter.apache.org/>.
- Apache.org. 2016.** Apache.org. *Apache JMeter.* [En línea] 2016. [Citado el: 22 de 11 de 2016.] https://httpd.apache.org/docs/trunk/es/new_features_2_2.html.
- Bareisa, E. 2015.** Development of case tools for software process improvement. [En línea] 2015. [Citado el: 10 de 11 de 2016.] <http://www.irem.ktu.lt/index.php/ITC/article/viewFile/11997/6672>.
- Bruno, Yuri. 2014.** Paginasweb.pe. *Importancia de un sistema de-gestion de contenidos CMS.* [En línea] 24 de 2 de 2014. [Citado el: 13 de 11 de 2016.] <http://paginasweb.pe/disenio-grafico/importancia-de-un-sistema-de-gestion-de-contenidos-cms/>.
- Cahueñas, Quishpe y Homero, Wilson. 2011.** Repositorio Institucional de la Universidad de las Fuerzas Armadas ESPE. *Desarrollo e implementacion del sistema de control para la emisión y recepción de la documentación de la Escuela Politécnica del Ejército Extensión Latacunga, aplicando herramientas Open Source y la metodología Extreme Programming.* [En línea] 3 de 2011. [Citado el: 27 de 8 de 2016.] <http://repositorio.espe.edu.ec:8080/bitstream/21000/4462/1/M-ESPEL-0052.pdf>.
- Campderrich, B. 2002.** *Ingeniería del software.* [En línea] 2002. [Citado el: 19 de 3 de 2017.] <https://books.google.com/cu/books?isbn=8484297934>. ISBN: 8484297934.
- Canós, José H. 2014.** *Metodologías Ágiles en el Desarrollo de Software.* 2014.
- Colombia, Universidad Nacional de. 2016.** Identidad: Universidad Nacional de Colombia. *Buenas prácticas para la web.* [En línea] 9 de 6 de 2016. [Citado el: 5 de 11 de 2016.] <http://identidad.unal.edu.co/guia-de-diseno-web/b-directrices-y-especificaciones/bp-buenas-practicas/>.

Compunet. 2016. Compunet. *Folleto Acunetix*. [En línea] 2016. [Citado el: 27 de 11 de 2016.] www.compunet.cl/pdf/Folleto-Acunetix.pdf.

Consulting, Agencia Marketing Web. 2013. Agencia Marketing Web Consulting. *Gestores de contenidos más utilizados actualmente*. [En línea] 2013. [Citado el: 20 de 12 de 2016.] <http://marketingwebconsulting.uma.es/gestores-de-contenidos-mas-utilizados-actualmente/>.

Cornejo, José Enrique González. 2008. Docirs. *UML*. [En línea] 1 de 2008. [Citado el: 18 de 11 de 2016.] <http://www.docirs.com/uml.htm>.

Devore, Jay L. 2010. *Probabilidades y Estadísticas para Ingeniería y Ciencias. Sexta Edición*. 2010.

Díaz, F. Javier, y otros. 2010. *Usando JMeter para pruebas de rendimiento*. La Plata : s.n., 2010.

Española, Diccionario de la lengua. 2005. WordReference. *Definición de publicación*. [En línea] 2005. [Citado el: 3 de 5 de 2017.] <http://www.wordreference.com/definicion/publicaci%C3%B3n>.

Española, Real Academia. 2017. Real Academia Española. *Concepto de módulo*. [En línea] 2017. [Citado el: 10 de 4 de 2017.] <http://dle.rae.es/srv/search?m=30&w=m%C3%B3dulo>.

—. **2015.** Real Academia Española. *Concepto de correo electrónico*. [En línea] 2015. [Citado el: 23 de 5 de 2017.] <http://lema.rae.es/dpd/srv/search?key=e-mail>.

Filippi, José Luis. 2009. Postgrado Facultad de Informática. *Método para la integración de TICs*. [En línea] 2009. [Citado el: 12 de 9 de 2016.] http://postgrado.info.unlp.edu.ar/Carreras/Magisters/Tecnologia_Informatica_Aplicada_en_Educacion/Tesis/Filippi.pdf.

García, Ana. 2014. Javiergarzas.com. *Tipos de pruebas*. [En línea] 2014. [Citado el: 5 de 4 de 2017.] <http://www.javiergarzas.com/2014/07/tipos-de-pruebas-10-min.html>.

Goñi, A., y otros. 2013. OpenSourceWare Universidad de País Vasco. *Realización de diagramas de frecuencia: capas de software y patrones WRASP*. [En línea] 2013. [Citado el: 15 de 4 de 2017.] https://ocw.ehu.eus/pluginfile.php/5196/mod_resource/content/1/3-3_Realizacion_Diagramas_Secuencia.pdf.

Hidalgo, Jorge. 2013. Ecomundo Centro de Estudios. *Taller de correos capítulo 5.* [En línea] 20 de 08 de 2013. [Citado el: 9 de 12 de 2016.]

https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=2&ved=0ahUKEwiRgfD_tNLTahWKgFQKHWGTBsAQFggrMAE&url=http%3A%2F%2Fgye.ecomundo.edu.ec%2Fdoc_aula_virtual_ecotec%2Fareas%2F2013E1%2FCOM170%2Falum%2F2012540964_4605_2013E1_COM170_fundamentos_pi.do.

Hispano, Drupal. 2011. Drupal Hispano. *Drupal.* [En línea] 2011. [Citado el: 15 de 11 de 2016.]

<http://drupal.org.es/caracteristicas>.

Hostdime. 2014. Hostdime. *Cómo publicar artículos para wordpress con el correo electrónico.* [En línea] 12 de 3 de 2014. [Citado el: 10 de 11 de 2016.] <http://blog.hostdime.com.co/como-publicar-articulos-para-wordpress-con-el-correo-electronico/>.

Jimenez, Aleman, Sosa, Yulio y Thomas y Jorge, Yoniel. 2014. *MÓDULO DE CONFIGURACIÓN PARA EL MECANISMO DE RASTREO DEL BUSCADOR ORIÓN.* La Habana : s.n., 2014.

Larman, Craig. 2016. *DSpace.* 2016.

León, Marcela. 2015. *¿Por qué son importantes las pruebas funcionales?* [En línea] 10 de 5 de 2015. [Citado el: 28 de 3 de 2017.] <http://www.kibernum.com/noticias/por-que-son-importantes-laspruebas-funcionales-2/>.

Lorenzo. 2012. Solo Joomla. *Publicar artículos por email y mms.* [En línea] 6 de 12 de 2012. [Citado el: 10 de 11 de 2016.] <http://www.solojoomla.com/publicar-articulos-por-email-y-mms.html>.

Márquez, Javier. 2013. WordPress. *Patrones GoF.* [En línea] 1 de 2013. [Citado el: 24 de 1 de 2017.] <https://infow.wordpress.com/category/patrones-de-disenogof/page/2/>.

Martínez, Mirna. 2015. Educación en línea. *Correo electrónico.* [En línea] 9 de 27 de 2015. [Citado el: 6 de 11 de 2016.] http://educacionenlineaequipomorado.blogspot.com/2015_09_01_archive.html.

Mifsud, Elvira. 2011. Observatorio Tecnológico. [En línea] 12 de 7 de 2011. [Citado el: 20 de 1 de 2017.]

- Morales , Arredondo y Azucena, Perla. 2009.** Monografías. *Servidores web*. [En línea] 2009. [Citado el: 2 de 12 de 2016.] <http://www.monografias.com/trabajos75/servidores-web/servidores-web/servidores-web.shtml>.
- Murcia, Universidad de. 2009.** Fundamentos del ingeniería de software. *Capítulo 5: Prueba del software*. [En línea] 5 de 6 de 2009. [Citado el: 6 de 5 de 2017.] <http://ocw.um.es/ingenierias/fundamentos-de-ingenieria-del-software/material-de-clase-1/capitulo05.pdf>.
- Nacional, Redacción. 2015.** Granma. *La sociedad en el camino de la informatización*. [En línea] 18 de 9 de 2015. [Citado el: 15 de 11 de 2016.] <http://www.granma.cu/cuba/2015-09-18/la-sociedad-en-el-camino-de-la-informatizacion>.
- Networks, Raiola. 2017.** Raiola Networks. *¿Que es Jetpack para WordPress? – Guía completa*. [En línea] 10 de 1 de 2017. [Citado el: 2 de 5 de 2017.] <https://raiolanetworks.es/blog/que-es-jetpack-para-wordpress-guia-completa/>.
- Oracle. 2014.** Sitio oficial del IDE NetBeans. *Características del IDE NetBeans*. [En línea] 2014. [Citado el: 23 de 10 de 2016.] <https://netbeans.org/features/index.html>.
- Paganotti, S. 2013.** *Designing Next Generation Web Projects with CSS3*. 2013. ISBN 978-1 -84969-326-4.
- PHP. 2012.** PHP. *What is PHP*. [En línea] 2012. [Citado el: 25 de 10 de 2016.] <http://php.net/manual/es/intro-what-is.php>.
- Pressman, R.S. 2001.** *Ingeniería de Software: Un enfoque práctico*. 5. s.l. : Mc Graw Hill., 2001. ISBN 0-07-285318-2..
- Pressman, Roger. 2007.** *Ingeniería del Software un enfoque práctico. Estrategia de pruebas*. 2007. ISBN: 97-0105-473-3..
- Raul, San Cristóbal de las Casas. 2016.** *Drupal como plataforma para redes de organizaciones*. 2016.
- Riemann, B. 2012.** Internet-didáctica. *Historia de Internet*. [En línea] 10 de 6 de 2012. [Citado el: 9 de 10 de 2016.] <http://www.internet-didactica.es/historia-internet/>.

Rodríguez Salas, Karla y Barboza Jiménez, Lucrecia. 2016. UNAM Universidad Nacional Autónoma de México. *Las TIC como apoyo al proceso de enseñanza-aprendizaje en Bibliotecología*. [En línea] 2016.

[Citado el: 24 de 3 de 2017.]

<http://iibi.unam.mx/publicaciones/280/tic%20educacion%20bibliotecologica%20las%20TICs%20Karla%20Rodriguez%20Salas.html>.

Rodríguez, Txema. 2012. GENBETA: dev. *Apache 2.4 HTTP Server disponible con mejoras de rendimiento adaptandose a los tiempos modernos*. [En línea] 28 de 2 de 2012. [Citado el: 25 de 10 de 2016.] <https://www.genbetadev.com/desarrollo-web/apache-http-server-2-4-disponible-con-mejoras-de-rendimiento-adaptandose-a-los-tiempos-modernos>.

Sánchez, S. 2015. Estrategia de soporte técnico para el proceso de migración a código abierto en los Organismos de la Administración Central del Estado. *Maestría*. La Habana : Universidad de las Ciencias Informáticas, 2015.

Sommerville, Ian. 2005. *Ingeniería del software 7ma edición*. 2005.

Thibaud, Cyril. 2006. *MySQL 5*. Barcelona : Amadeu Brugués, 2006. ISBN:2-7460-3069-1.

Torets, Jose Manuel López. 2016. V&V Quality. *Pruebas de seguridad*. [En línea] 2016. [Citado el: 6 de 4 de 2017.] <http://vyvquality.com/pruebas-seguridad/>.

Verma, Pandey y Bansal. 2014. *Develop Framework for Selecting Best Software Development*. 2014.

Vojacek, Alex. 2012. Tecnogaming. *Web Servers: NGINX vs Apache vs Lighttpd*. [En línea] 24 de 4 de 2012. [Citado el: 13 de 9 de 2016.] <https://www.tecnogaming.com/2012/04/web-servers-nginx-vs-apache-vs-lighttpd/>.

www.uci.cu. 2017. www.uci.cu. www.uci.cu/mision. [En línea] 2017. [Citado el: 12 de 3 de 2017.] www.uci.cu/universidad/mision.

Anexos

Anexo1. Historias de usuario

Tabla 11. H.U. Verificar datos de la cuenta.

Número: 6	Nombre del requisito: Verificar datos de la cuenta.
Programador: Bárbara B. Somoza Leal	Iteración Asignada: Iteración 1
Prioridad: Alta	Tiempo Estimado: ½ día
Riesgo en Desarrollo: 1.-Atraso en la entrega del diseño de interfaz del módulo. 2.-Falla del fluido eléctrico. 3.-Fallo en la red.	Tiempo Real: ½ día
Descripción: Al crearse una nueva cuenta el sistema verifica si existe conexión con el correo electrónico introducido por el usuario y si la contraseña coincide con la de ese correo.	
Observaciones: 1- Si el sistema ejecuta correctamente la petición se mostrará un mensaje notificando que se han verificado satisfactoriamente los datos de la cuenta.	
Prototipo de interfaz: No aplica	

Tabla 12. H.U. Revisar los correos electrónicos

Número: 10	Nombre del requisito: Revisar los correos electrónicos.
Programador: Bárbara B. Somoza Leal	Iteración Asignada: Iteración 1
Prioridad: Alta	Tiempo Estimado: ½ día
Riesgo en Desarrollo: 1.-Atraso en la entrega del diseño de interfaz del módulo. 2.-Falla del fluido eléctrico. 3.-Fallo en la red.	Tiempo Real: ½ día
<p>Descripción:</p> <p>El sistema accede a través de la configuración previamente realizada de la cuenta al correo electrónico especificado, verifica si alguno de ellos cumple con las configuraciones establecidas para publicarlo en el sitio web y notifica la cantidad de correos existentes, así como los que se pueden publicar en el sitio web. Si el sistema no puede ejecutar correctamente la petición se mostrará un mensaje notificando que no se ha revisado satisfactoriamente el correo electrónico .</p>	

Observaciones:

1- Si el sistema ejecuta correctamente la petición se mostrará un mensaje notificando que se ha revisado satisfactoriamente el correo electrónico.

2- Si el sistema no puede ejecutar correctamente la petición se mostrará un mensaje notificando que no se ha revisado satisfactoriamente el correo electrónico .

Prototipo de interfaz: No aplica.

Anexo2. Casos de prueba

Tabla 13. Caso de prueba 2 Editar Cuenta

Caso de Prueba 2	
Código: CDPHU2	Nombre: Editar cuenta.
Nombre del probador: Bárbara Beatriz Somoza Leal.	
Descripción: Prueba a la funcionalidad editar cuenta.	
Condiciones de Ejecución: El usuario debe estar autenticado y tener los permisos necesarios.	
Entrada / Pasos de ejecución: <ul style="list-style-type: none">• El usuario accede al sistema.• El usuario da <i>click</i> en el enlace Publishmail encontrado en el menú de administración.• El sistema mostrará la página principal de configuración del módulo.• El usuario dará <i>click</i> al enlace Cuenta el cual mostrará la página de configuración de las cuentas del módulo.• El usuario seleccionará la opción Editar que le permitirá acceder a la ventana con los campos requeridos para editar la cuenta.	

- Después de llenar los campos requeridos el usuario dará clic en el botón Salvar encontrado al final de la ventana.
- El sistema mostrará un mensaje confirmando que se ha editado correctamente la cuenta.

La entrada consta de la introducción de los datos en los siguientes campos:

Correos autenticados: usuario@ejemplo.com

Usuario: nombre_usuario

Dirección de correo: usuario@ejemplo.com

Contraseña: 123admin.

Protocolo: IMAP

Dominio: imap.ejemplo.com

Puerto: 143

Comandos extras: /novalidate-cert

Escenarios:	Resultados Esperados	Evaluación de la Prueba:
EC1. El usuario introduce correctamente los datos.	El sistema edita la cuenta y muestra el siguiente mensaje de confirmación: ``Se ha creado satisfactoriamente la cuenta``.	Satisfactoria.
EC2. El usuario introduce datos usados en otra cuenta.	El sistema no edita la cuenta y muestra el siguiente mensaje de error: ``Ha introducido datos que ya están en uso, verifique los campos de la cuenta``.	Satisfactoria.
EC3. El usuario deja uno o varios campos vacíos.	El sistema no edita la cuenta y muestra el siguiente mensaje de	Satisfactoria.

	error: ``Ha dejado campos vacíos, verifique los campos de la cuenta ``.	
EC4. El usuario introduce campos incorrectos.	El sistema no edita la cuenta y muestra el siguiente mensaje de error: ``Ha introducido campos incorrectos, verifique los campos de la cuenta ``.	Satisfactoria.

Tabla 14. Caso de prueba 5 Clonar Cuenta

Caso de Prueba 5	
Código: CDPHU5	Nombre: Clonar cuenta.
Nombre del probador: Bárbara Beatriz Somoza Leal.	
Descripción: Prueba a la funcionalidad Clonar cuenta.	
Condiciones de Ejecución: El usuario debe estar autenticado y tener los permisos necesarios.	
Entrada / Pasos de ejecución: <ul style="list-style-type: none"> • El usuario accede al sistema. • El usuario da clic en el enlace Publishmail encontrado en el menú de administración. • El sistema mostrará la página principal de configuración del módulo. • El usuario dará clic al enlace Cuenta el cual mostrará la página de configuración de las cuentas. • El usuario seleccionará la opción Clonar que se encuentra en el menú desplegable a la derecha de la cuenta previamente creada, el cual le permitirá clonar la cuenta y le permitirá acceder a la ventana con los campos requeridos para editarla. • Después de llenar los campos requeridos el usuario dará clic en el botón Salvar encontrado al final de la ventana. • El sistema mostrará un mensaje confirmando que se ha clonado correctamente la cuenta. 	

Escenarios:	Resultados Esperados	Evaluación de la Prueba:
EC1. El usuario selecciona la opción Clonar y edita correctamente los datos.	El sistema muestra la ventana de edición de los campos de la cuenta y si el usuario edita correctamente los datos se mostrará el siguiente mensaje de confirmación: ``Se ha clonado satisfactoriamente la cuenta``.	Satisfactoria.
EC2. El usuario selecciona la opción Clonar y no edita correctamente los datos.	El sistema muestra la ventana de edición de los campos de la cuenta y si el usuario no edita correctamente los datos se mostrará el siguiente mensaje: `` No se ha clonado satisfactoriamente la cuenta``.	Satisfactoria.