

**Universidad de las Ciencias Informáticas**

**Facultad 3**



**Sistema para la edición y publicación electrónica de la  
Gaceta Oficial de la República de Cuba**



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

**Autor(es):** Amarilys de la Caridad Ivonet Quiala.

Ledier Hernández Martínez.

**Tutor:** MsC. Yarina Amoroso Fernández.

**Co-Tutor:** Ing. Daniel Varona Cordero.

*La Habana, 2014.*

*“Año 56 de la Revolución”*

Albert Einstein

Hay una fuerza  
motriz más poderosa  
que el vapor,  
la electricidad y  
la energía atómica:  
la voluntad



## Declaración de autoría

Declaramos que somos los únicos autores de la presente tesis y reconocemos a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

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

\_\_\_\_\_

Amarilys de la Caridad Ivonet Quiala

Firma del Autor

\_\_\_\_\_

MsC. Yarina Amoroso Fernández

Firma del Tutor

\_\_\_\_\_

Ledier Hernández Martínez

Firma del Autor

\_\_\_\_\_

Ing. Daniel Varona Cordero

Firma del Co-Tutor

### *Amarifys*

A mis **padres** por estar siempre que los necesité, por brindarme su apoyo y su confianza, por preocuparse siempre por mí, por darme tanto ánimo y porque sin ustedes no estuviera aquí.

A mi **hermano** y a Adri por darme tanto apoyo, por esforzarse para que nunca me faltara nada y pudiera estudiar sin ninguna preocupación.

A mis **abuelos**, en especial a mi abuela Migdalia por preocuparse tanto por mi cada vez tenía alguna prueba.

A mi **familia de emos** de aquí de la UCI, Lili, Lety, Ibe y Oni, de verdad muchas gracias por nunca dejarme sola y apoyarme tanto, por enseñarme el valor de la verdadera amistad, sin ustedes nunca hubiera llegado hasta aquí, gracias a ustedes se me hizo este camino un poquito más fácil de recorrer, gracias por compartir cada instante de su vida conmigo, los voy a extrañar mucho mis amigos.

A mi **novio** Anthuan por sacrificarse y brindarme su apoyo cuando más lo he necesitado, por sentirse orgulloso de mí, por ser compañero y amigo y por brindarme su amor que tanto bien me ha hecho.

A mi **amigo** el gorra que aunque ya no esté aquí se comportó conmigo como un hermano.

A mis **tíos**, mis primos por su preocupación constante y por el apoyo brindado.

A mi **compañero** de tesis Ledier por todo el tiempo que hemos pasado juntos.

A todos mis **compañeros** de aula que me han acompañado estos años de estudios, por los buenos y malos momentos pasados juntos, por el aprecio y la estima de todos, nunca los olvidaré estén donde estén, siempre los tendré presentes.

A mis **amigos** Alexander, Yesenia, Carlos Iván, Dayán, Carlos y Becky porque a pesar de la distancia nunca se han olvidado de la amistad que existe entre nosotros.

A mi **tutora** por brindarnos todo su apoyo para poder cumplir este sueño.

A **todos los profesores** que contribuyeron en mi formación como profesional.

A **todos** por nunca perder la fe y la confianza en mí, muchas gracias.

## *Ledier*

Quiero agradecer ante todo a mi **mamá** por darme la vida y todo su amor. Por ser la principal guía y educadora de mi vida. Por enseñarme que en la vida todo se gana con esfuerzo y sacrificio.

A mi **papá** que por cosas de la vida ya no está entre nosotros le agradezco por haberme dado la oportunidad de existir y que Dios lo proteja siempre.

A mi **padrastró** por ser para mí un padre desde que tengo 2 años de vida y quererme como un hijo.

A mis **abuelos** y en especial a Lidia y Gallego por formar una parte importante de mi vida, por consentirme todo el tiempo y por la fuerza de voluntad que me han inspirado siempre a ser una mejor persona.

A mis **tíos** y en especial a Carmen y Lázaro por ser para mí como mis padres.

A mis **hermanos** y en especial a Lester por siempre estar a mi lado.

A mi **prima** Arianna por ser como una hermana para mí y quererme con la vida.

A mi **familia** que de una manera u otra han colaborado para que este sueño se haga realidad.

A mis **amigas** Maidelys, Danay por ser como una hermana para mí.

A mi **grupo** de clases por soportarme y ser tan unido y pasar junto todos estos años de estrés.

A mis **amigos** más querido Jairo, Ramón, Ernesto, Aliuska, Osiel, Lismey, Arletis, Alberto, Luis Enrique, Yaniel, Yanisley, Alejandro, Dalili, Yanet, Victoria, Las Mellizas, Renier, Rafael, José Carlos, Solanch, Paumier, Pedro y Andy, Dayana Francia, por compartir conmigo momentos especiales, por ser también para mí un pedacito importante de mi corazón, por ser de las personas que nunca olvidaré.

A los **profesores** Hugo Valgas, Dariela, Maye, Naryana, Lien, Odette, Yeni Calderin.

A mi **tutora** Yarina por todo su apoyo y ayuda a realizar este sueño y ser una persona que admiro y nunca olvidaré.

A nuestro **tribunal** por todo su compromiso, por cada acertado consejo.

A mi **compañera** de tesis por compartir conmigo todo este tiempo de dedicación.

A todo el que me preguntó alguna vez: "y la tesis... ¿cómo va?"

*De Amarilys*

*Dedico el resultado de este trabajo y de toda mi vida como estudiante, a mis padres, por sus infinitos sacrificios, por sus palabras de aliento en los momentos más difíciles, porque gracias a ellos estoy aquí, los quiero mucho.*

*A familia y amigos en general porque al cumplir este sueño, cumplo uno de los suyos.*

*De Ledier*

*Dedico esta tesis a mi madre, la persona que más amo en mi vida. Por apoyarme en toda mi trayectoria estudiantil así como las decisiones que he tomado en mi vida.*

## Resumen

La presente investigación describe la realización de un sistema para la edición y publicación electrónica de la Gaceta Oficial de la República de Cuba para el Departamento de Dirección de Informática Jurídica del Ministerio de Justicia desarrollado en el Centro de Gobierno Electrónico, basándose en los principios jurídicos para la publicidad normativa. El desarrollo de este sistema permite a los ciudadanos visualizar o descargar todas las leyes, decretos, resoluciones, reglamentos y demás disposiciones generales de los órganos nacionales del estado, entre otras opciones. La solución propuesta contribuye a corregir deficiencias presentadas por las soluciones que le anteceden, destacándose dentro de ellas que los procesos de edición y publicación de la Gaceta Oficial de la República de Cuba se realizan de forma manual. Se incorporan además, nuevas funcionalidades y servicios de utilidad para los usuarios, como suscribirse al sistema, realizar búsquedas avanzadas y el envío de comentarios. Para ello se utilizó como metodología de desarrollo de software Extreme Programming. La implementación del sistema fue basada en la utilización de Drupal como sistema gestor de contenido y marco de trabajo para el desarrollo, PostgreSQL como sistema gestor de base de datos y NetBeans como entorno integrado de desarrollo lo que permitió obtener un sistema de fácil manejo y de alta calidad. Se obtuvo como resultado con la implementación del sistema web la optimización de los procesos de edición y publicación electrónica. La propuesta de solución cumple con las tendencias y los estándares actuales del desarrollo de sistemas web.

**PALABRAS CLAVES:** Drupal, Gaceta Oficial, Informática Jurídica, Ministerio de Justicia, Sistema Web.

# Índice de contenido

<b>Introducción</b> .....	1
<b>Estructura del Documento</b> .....	4
<b>Capítulo 1: Fundamentación teórica de la investigación</b> .....	5
1.1. Conceptos asociados al dominio del problema .....	5
1.1.1 Publicidad del Derecho .....	5
1.1.2 Gaceta Oficial (GO) .....	6
1.2. Estado del arte.....	7
1.2.1 Publicación electrónica .....	7
1.2.2 Soluciones existentes a nivel internacional y nacional .....	9
1.3 Metodología de desarrollo de Software.....	10
1.3.1 Extreme Programming (XP) .....	12
1.4 Sistema de Gestión de Contenido (CMS) .....	14
1.4.1 WordPress.....	14
1.4.2 Drupal.....	14
1.4.3 Joomla.....	16
1.4.4 Selección del sistema de gestión de contenido a utilizar.....	16
1.5 Sistema Gestor de Base de Datos (SGBD).....	16
1.5.1 PostgreSQL.....	17
1.5.2 MySQL.....	18
1.5.3 SQLite.....	19
1.5.4 Selección del sistema de gestión de base de datos a utilizar .....	19
1.6 Tecnología y herramientas utilizadas en la implementación del sistema web .....	19
1.6.1 Lenguaje de Modelado .....	19
1.6.2 Lenguaje del lado del cliente.....	21

1.6.3	Lenguaje del lado del servidor .....	22
1.6.4	Herramienta de Modelado.....	22
1.6.5	Entorno de Desarrollo Integrado (IDE).....	23
1.6.6	Herramienta de prueba de software.....	23
1.6.7	Servidor Web.....	24
1.7	Conclusiones parciales .....	24
<b>Capítulo 2: Planificación, diseño .....</b>		<b>25</b>
2.1.	Descripción del sistema .....	25
2.2.	Modelado del negocio.....	26
2.3.	Planificación.....	28
2.3.1	Historias de usuarios detectadas .....	28
2.3.2	Requisitos del sistema .....	30
2.3.3	Técnicas de validación de requisitos.....	34
2.3.4	Métricas usadas para la validación de la calidad de los requisitos .....	34
2.3.5	Estimación de esfuerzo por historias de usuarios .....	36
2.3.6	Plan de iteraciones .....	37
2.3.7	Plan de duración de iteraciones .....	38
2.3.8	Plan de entrega .....	38
2.4.	Arquitectura del sistema.....	39
2.5.	Patrones de diseño .....	41
2.6.	Seguridad de la solución propuesta .....	42
2.7.	Diseño .....	43
2.7.1	Tarjetas CRC (Cargos o Clases, Responsabilidad y Colaboración).....	43
2.7.2	Diagrama de paquetes.....	44
2.8.	Conclusiones parciales .....	45
<b>Capítulo 3: Implementación y Validación de la propuesta solución .....</b>		<b>46</b>

3.1. Diagrama de Componentes .....	46
3.2. Tareas de ingeniería .....	47
3.3. Estándares de código .....	47
3.4. Diagrama de Despliegue.....	49
3.5. Validación .....	50
3.6. Definición de la Estrategia de Pruebas .....	50
3.7. Validación de la Investigación.....	57
3.8. Pantallas de la aplicación.....	58
3.9. Conclusiones parciales .....	59
<b>Conclusiones generales .....</b>	<b>60</b>
<b>Recomendaciones .....</b>	<b>61</b>
<b>Referencias Bibliográficas .....</b>	<b>62</b>
<b>Bibliografía consultada .....</b>	<b>66</b>
<b>Glosario de términos .....</b>	<b>69</b>

## Índice de figuras

FIGURA 1: REPRESENTACIÓN DEL PROCESO DE LEGISLACIÓN EN CUBA. _____	6
FIGURA 2: VOCABULARIO DE UML (JACOBSON, Y OTROS, 2000). _____	20
FIGURA 3: DIAGRAMA DE PROCESO DE NEGOCIO. _____	27
FIGURA 4: ARQUITECTURA DE DRUPAL (RODRÍGUEZ, 2012). _____	40
FIGURA 5: DIAGRAMA DE PAQUETES DEL SISTEMA. _____	44
FIGURA 6: DIAGRAMA DE COMPONENTES GENERAL DEL SISTEMA WEB PARA LA GACETA OFICIAL. ____	46
FIGURA 7: MODELO DE DESPLIEGUE. _____	49
FIGURA 8: ESTRATEGIA DE PRUEBAS. _____	51
FIGURA 9: RESULTADOS DE LA PRUEBA DE ACEPTACIÓN. _____	54
FIGURA 10: RESULTADOS DE LAS PRUEBA DE SEGURIDAD. _____	55
FIGURA 11: RESULTADOS DE LA PRUEBA DE USABILIDAD. _____	57
FIGURA 12: PANTALLA DE LA PORTADA PRINCIPAL DEL SISTEMA. _____	58
FIGURA 13: PANTALLA DE LA FUNCIONALIDAD VISUALIZAR LA GO. _____	59

## ÍNDICE DE TABLAS

TABLA 1: DIFERENCIA ENTRE METODOLOGÍAS ÁGILES Y NO ÁGILES. _____	11
TABLA 2: METODOLOGÍAS ÁGILES MÁS UTILIZADAS. _____	12
TABLA 3: HISTORIA DE USUARIO: GESTIONAR GO. _____	29
TABLA 4: TABLA DE ESTIMACIÓN DE ESFUERZO POR HU. _____	37
TABLA 5: PLAN DE DURACIÓN DE LAS ITERACIONES. _____	38
TABLA 6: PLAN DE ENTREGA. _____	39
TABLA 7: RESPONSABILIDAD Y COLABORACIÓN DEL MÓDULO GESTIONARGO. _____	43
TABLA 8: PRUEBA DE ACEPTACIÓN DE LA HU REALIZAR BÚSQUEDA AVANZADA DE LA GO. _____	53
TABLA 9: RESULTADOS DE LA HERRAMIENTA WEBSECURIFY. _____	55
TABLA 10: RESULTADOS PRUEBAS DE RENDIMIENTO. _____	56

## Introducción

El continuo avance de las tecnologías de la información y las comunicaciones ha tenido un fuerte impacto en la manera de generar y difundir el conocimiento e información a los ciudadanos. En la última década se ha producido un importante fenómeno editorial con el cambio del medio impreso al formato electrónico y con el surgimiento de nuevas publicaciones en línea, lo cual ha traído como consecuencia el auge de las publicaciones electrónicas. Si bien no es posible suponer que el medio impreso eventualmente desaparecerá del mundo de las publicaciones, no se puede negar el avance decisivo de las publicaciones electrónicas a nivel internacional.

El Ministerio de Justicia (MINJUS) no ha estado exento de estas transformaciones en la actividad editorial. Los cambios producidos han estado asociados fundamentalmente a la introducción de las tecnologías para el tratamiento de la información, almacenamiento y edición de publicaciones en formato electrónico, al desarrollo de las redes de telecomunicaciones locales y los servicios a ellas asociadas, así como, el incremento de la explotación de los servicios de correo electrónico e Internet.

Estos cambios han representado una nueva etapa de desarrollo orientada hacia el mayor aprovechamiento de las capacidades tecnológicas con que se cuenta, ofrecer servicios y productos de alto valor añadido con mayor eficiencia, calidad y con una adecuada orientación a las necesidades de los usuarios.

En la actualidad, en el MINJUS de la República de Cuba se desarrollan procesos relacionados a la responsabilidad que tiene el estado de dar a conocer la legislación que se promulga. Los emisores oficiales de disposiciones jurídicas remiten al MINJUS los actos normativos para su evaluación e inclusión en la Gaceta Oficial (GO) correspondiente como órgano de publicidad normativa. La GO es distribuida en tres formatos fundamentales, el tradicional en papel, a través del correo electrónico o digital y es difundida en el sitio web oficial de la GO de la República de Cuba.

El MINJUS cuenta actualmente con un sitio web que se encarga de divulgar los documentos de la GO que ya han sido aprobados en la Asamblea Nacional del Poder Popular, pero este presenta deficiencias que dificultan un fácil acceso y manejo de las leyes. Este sitio no maneja correctamente los procesos de edición y publicación por lo que dificulta un rápido acceso a las

leyes por parte de los usuarios, por esta razón, la ciudadanía emplea gran parte de su tiempo dirigiéndose a los centros directamente vinculados con las leyes para informarse sobre estas. Además al acceder a este sitio se puede observar como proceso de edición y publicación, el escaneo del último número de la GO dificultándose su lectura en línea y la búsqueda de una ley en específico por los usuarios en red. Otros problemas que presenta el sitio es que no permite una correcta administración, no brinda la posibilidad de suscribirse para recibir los últimos números de la gaceta que hayan sido subidos al sitio, no deja descargar ni imprimir en formato PDF los documentos de la GO, por lo que se puede decir que este sitio no cumple de forma correcta con los objetivos para los que fue creado.

El proceso de edición y publicación de la GO de la República de Cuba asociado al procedimiento de evaluación técnica que realiza el MINJUS como máximo órgano de asesoría del Estado y el Gobierno en materia legislativa se continúa haciendo de forma manual.

La situación anteriormente descrita, unida a la necesidad del uso adecuado de las herramientas y métodos necesarios para modernizar el proceso de edición y publicación de la GO de la República de Cuba, permitió identificar como **problema a resolver**: Cómo mejorar el canal existente de comunicación y servicios a la comunidad de usuarios que intervienen en los procesos de edición y publicación de la Gaceta Oficial de la República de Cuba.

Para solucionar el problema se define como **objeto de estudio de la investigación**: Procesos de edición y publicación de la Gaceta Oficial.

Se concibe como **objetivo de la investigación**, desarrollar un sistema soportado en tecnología web para mejorar el canal de información y los servicios a la comunidad de usuarios que intervienen en los procesos de edición y publicación de la Gaceta Oficial de la República de Cuba y suplir las necesidades de dichos procesos.

De la relación entre el problema, el objeto de estudio y el objetivo se identifica como **campo de acción**, la gestión de documentos digitales para la edición y publicación de la Gaceta Oficial de República de Cuba.

Para dar solución al problema se plantea como **idea a defender**, un sistema soportado en tecnología web fácil de utilizar y basado en nuevas tecnologías, permitirá mejorar el canal existente de comunicación y servicios a la comunidad de usuarios que intervienen en los procesos de edición y publicación de la Gaceta Oficial de la República de Cuba, así como satisfacer las necesidades de dichos procesos.

Para darle cumplimiento al objetivo general se plantean como **objetivos específicos**:

1. Elaborar el marco teórico de la investigación.
2. Confeccionar la propuesta de solución.
3. Validar la solución propuesta.

Para el cumplimiento de los objetivos se plantean las siguientes **tareas de investigación** a desarrollar:

1. Fundamentación teórico-metodológica sobre los procesos de edición y publicación electrónica.
2. Fundamentación de la información de experiencias desarrolladas en Iberoamérica.
3. Fundamentación de las tecnologías, herramientas, metodología y lenguaje necesarios para el desarrollo del producto.
4. Selección, aplicación y documentación de un estándar de codificación adecuado.
5. Levantamiento de los requisitos del software.
6. Modelación del ciclo del software.
7. Desarrollo de los artefactos teniendo en cuenta la metodología a utilizar.
8. Identificación y fundamentación de los patrones arquitectónicos a utilizar.
9. Validación de la propuesta del prototipo funcional del sistema.
10. Presentación de los resultados.
11. Introducción de los resultados en la práctica.

Para el desarrollo de este trabajo fueron utilizados los siguientes **métodos de investigación**

### **Métodos teóricos:**

**Histórico–Lógico:** Es utilizado debido a la necesidad de realizar un estudio acerca de las soluciones de sistemas de gestión de documentos legales que existen en el mundo, así como las tecnologías y estándares utilizados para su desarrollo.

**Analítico–Sintético:** Se define con el objetivo de analizar los diversos documentos relacionados con el proceso de edición y publicación electrónica y las metodologías existentes para la construcción de los artefactos necesarios así como para realizar una síntesis de los mismos.

**Modelación:** Se emplea para mostrar los diferentes diagramas y componentes que se construyen como resultado del proceso de Ingeniería de Software.

## **Métodos empíricos:**

**Entrevista:** Se realizan a líderes de proyectos para recopilar información referente al proceso de desarrollo en el mundo, en Cuba, así como para identificar las herramientas óptimas destinadas a la construcción de dichos sistemas.

**Medición:** Este método se tiene en cuenta con las pruebas que se le realizan al software y las métricas.

## **Estructura del Documento**

El trabajo de diploma cuyo resultado está dado en un Sistema web para la edición y publicación electrónica de la GO de la República de Cuba, incluyendo toda la documentación del mismo, se estructura en tres capítulos, que son descritos a continuación:

**Capítulo 1:** Fundamentación de la Investigación: Son expuestos los principales conceptos que contribuyen al mejor entendimiento del problema en cuestión, se especifican detalladamente todos los argumentos que esclarecen el objeto de estudio caracterizando los sistemas de gestión de documentos legales. Se presenta el estado del arte de las herramientas implicadas en el objeto de estudio y el conjunto de las técnicas, herramientas, lenguajes y metodología involucrados en el desarrollo de la propuesta y se realiza la justificación de las seleccionadas para la solución del problema.

**Capítulo 2:** Planificación, diseño: Se definen los principales requisitos de la propuesta de solución, las historias de usuarios y el refinamiento del análisis y el diseño. Se describen los módulos que se utilizan en la implementación del sistema y las relaciones entre ellos. Se define además la arquitectura del sistema, haciendo énfasis en los patrones arquitectónicos y de diseño utilizados. Se realiza el diseño del sistema mediante los distintos artefactos propuestos por la metodología de desarrollo de software seleccionada.

**Capítulo 3:** Implementación y Validación: Se describen los principales módulos del sistema, así como los artefactos correspondientes a la fase de implementación de la metodología seleccionada. También se emplean los tipos de casos de prueba adecuados para validar la aplicación que permiten detectar y corregir los posibles errores a través de los resultados obtenidos en cada una de las iteraciones.

Para finalizar se presentan las conclusiones, recomendaciones, referencias bibliográficas, bibliografía, glosario de términos y el conjunto de anexos para un mejor entendimiento de lo expuesto a lo largo de la investigación.

## **Capítulo 1: Fundamentación teórica de la investigación**

El enfoque de este capítulo es precisamente abordar en el diseño teórico de la investigación, iniciando con algunos conceptos asociados al tema en cuestión. Además se realiza un estudio de los principales sistemas de gestión de documentos legales existentes, así como conceptos asociados a estos para ayudar a una mejor comprensión del problema. Se describen las principales características de las herramientas y tecnologías que se utilizarán en el desarrollo de la solución.

### **1.1. Conceptos asociados al dominio del problema**

Con el objetivo de que el lector pueda tener una comprensión mayor de los temas que se relacionan en el capítulo, se describen detalladamente a continuación un grupo de conceptos asociados al dominio del problema, entre los que se destacan: Proceso de elaboración legislativa, Gaceta Oficial, entre otros.

#### **1.1.1 Publicidad del Derecho**

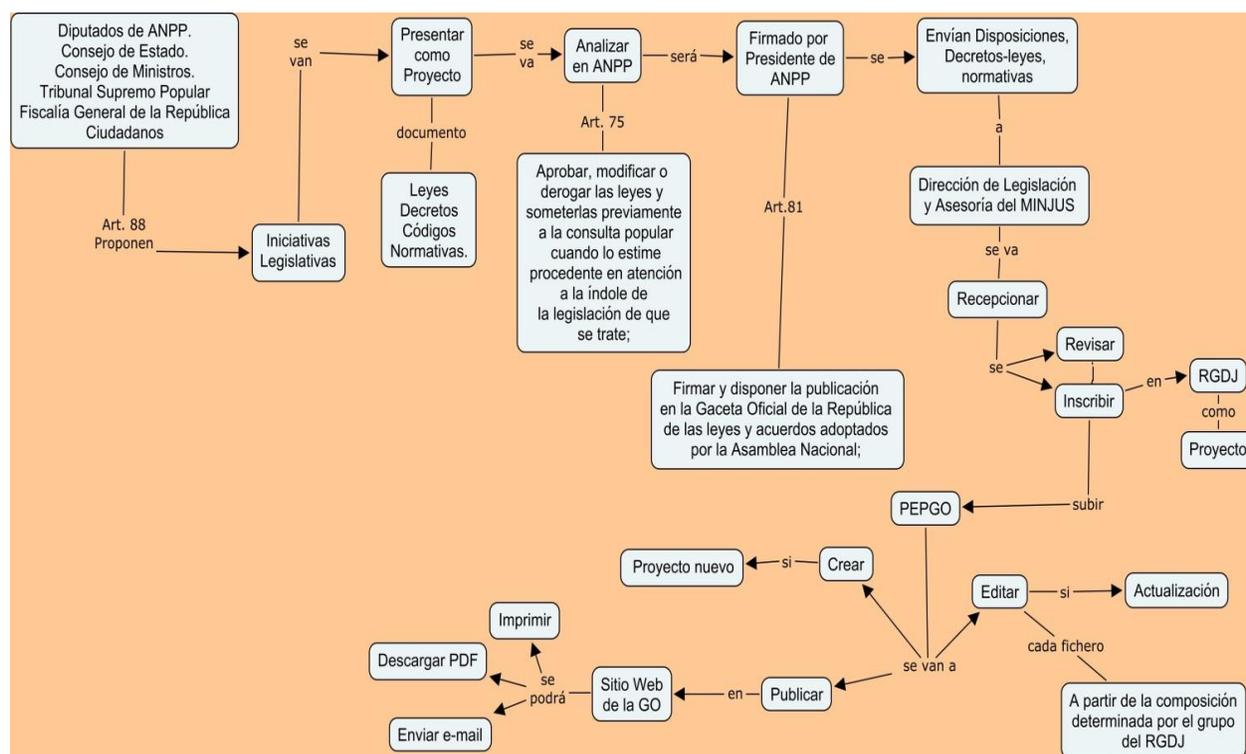
Conocer el Derecho constituye un derecho ciudadano y la vez se erige en una responsabilidad del Estado de darlo a conocer, es ello una de las razones que en los textos constitucionales se establezcan preceptos que establecen las normas fundamentales que rigen para la Publicidad de las normas (Amoroso, 2000).

La elaboración de principios y reglas a partir de análisis casuísticos de los problemas de la vida social, es el procedimiento lógico que posibilita el logro del "tipo" que la generalidad de la regulación exige, con carácter obligatorio, en las relaciones sociales. Dichas disposiciones jurídicas son al mismo tiempo las que establecen el referido orden que se va formando, a través de un proceso estructurado de funciones y creaciones normativas, sin las que desde luego, la fuerza vinculatoria de sus normas deba fundamentarse en la estructura normativa del Derecho mismo.

No pocas veces se ha confundido la promulgación de la ley con su publicación, siendo sin embargo dos instantes diferentes del proceso de creación legislativa. La publicación fija el inicio del conocimiento de la Ley por los destinatarios de su disciplina. Aunque el conocimiento de la ley y publicidad de la misma son dos cosas diferentes, representan el problema de determinar si la misma resulta obligatoria para el que conoce un texto durante el período que media entre la promulgación y la publicación. Además, la publicación puede no determinar el inicio de la

entrada en vigor de la ley y por ello no se hace obligatoria su observancia, cuando la propia ley aplaza el comienzo de su entrada en vigor. Si bien en el pasado las dificultades de publicación jurídica incidía hasta en la vigencia espacial de las normas por las distancias de los territorios donde debían aplicarse, hoy el desarrollo de la ciencia de la Informática, de las técnicas, medios de información masiva y de su aplicación al derecho debe llevar al campo de la vigencia temporal y espacial del derecho conscientes modificaciones (Amoroso, 2000).

A continuación en la Figura 1 se muestra el proceso de legislación de las leyes en Cuba.



**Figura 1: Representación del proceso de legislación en Cuba.**

### 1.1.2 Gaceta Oficial (GO)

El diario oficial, boletín oficial, gaceta oficial, registro oficial o periódico oficial es el medio de comunicación escrito que un Estado, o una organización internacional o regional, utiliza para publicar sus normas jurídicas, tales como: constituciones, tratados, leyes, decretos y reglamentos y otros actos de naturaleza pública. Una vez que las normas o decisiones han sido aprobadas o sancionadas por el o los respectivos órganos de los poderes del Estado (poder legislativo, poder ejecutivo y poder judicial), deben ser promulgadas y publicadas para que tengan efectos jurídicos y por tanto sean acatadas y reconocidas como parte del ordenamiento jurídico del Estado. Esto con base en el principio de que la ley debe ser conocida por el público

para que sea legítima. En algunas situaciones, particularmente en América Latina, cuando la norma aprobada requiere reserva por referirse a temas de la defensa nacional o labores de inteligencia, se publica una separata de circulación restringida o reservada. En esos casos se habla de leyes secretas (Amoroso, 2000).

### **1.2. Estado del arte**

Es muy útil que antes del desarrollo de cualquier investigación se realice un estudio profundo y valorativo de los antecedentes de la misma tanto a nivel internacional como nacional, por lo cual a continuación se presenta el correspondiente estudio relacionado con los proceso de edición y publicación electrónica de la GO.

#### **1.2.1 Publicación electrónica**

##### **¿Qué se entiende por publicación electrónica?**

A pesar de la frecuencia con que se escucha y se emplea el término de publicación electrónica y que de una forma cotidiana se hace uso de este tipo de publicación, las respuestas sobre qué se entiende por publicación electrónica son diversas.

Algunos autores como Brailstord en su artículo “Electronic publishing. A Journal and its production” plantean que el término electronic publishing (publicación electrónica) trasciende la “oficina sin papeles” o la pura transferencia y diseminación electrónica de información mediante redes y que incluye todos los métodos asociados a la producción de documentos con el uso de computadoras (Brailstord, y otros, 1989).

La Enciclopedia electrónica plantea que aunque a veces suele describirse como la aplicación de la computación al proceso de edición de la publicación tradicional, el término de publicación electrónica se refiere más precisamente al almacenamiento y distribución de información mediante el uso de la comunicación electrónica (The Electronic Encyclopedia, 1988).

Según Hechevarría (Hechevarría, 1997) existen varios enfoques sobre el concepto de publicaciones electrónicas los cuales se exponen a continuación:

- ✓ Es la forma digital de la publicación impresa.
- ✓ Es aquella que en su proceso de producción y edición incluye el uso de las computadoras.
- ✓ Es la oficina sin papeles. Todos los documentos están en formato electrónico.
- ✓ Es toda publicación soportada en medios magnéticos.

- ✓ Es la que se disemina a través de redes.
- ✓ Es toda publicación que se caracteriza por su capacidad multimedia y por almacenarse y distribuirse de forma electrónica.

Otro elemento importante (Espinosa, y otros, 2002) a tener en cuenta es que las **publicaciones electrónicas permitan:**

- ✓ La interacción directa con el usuario como si estuviera efectuándose una conversación.
- ✓ El acceso rápido y directo a través de enlaces sin necesidad de llevar una secuencia lineal.
- ✓ Hacer uso de la información, poderla imprimir y crear un nuevo documento a su medida.
- ✓ Prestaciones audiovisuales en las que se combinen textos, imágenes (estáticas y animadas) y sonidos.
- ✓ Almacenar y conservar la información en soportes magnéticos y ópticos con una gran capacidad de memoria, pudiéndose almacenar en poco espacio físico grandes volúmenes de información incluyendo imágenes, vídeo y sonido.
- ✓ La inserción de otros elementos que proporcionan valor añadido a la información.

**Ventajas de las publicaciones electrónicas** (Aguar, 2003).

- ✓ **Economía:** No genera gastos de impresión y encuadernación, ellos son sumamente caros en el caso de las revistas y libros impresos.
- ✓ **Bajos costos de almacenamiento:** El original se guarda en un disco, CD o disco ZIP. Se pueden crear un número infinito de copias en cualquier momento.
- ✓ **Facilidad de envío (inmediatez):** Se puede distribuir, enviar y recibir en forma casi inmediata a través de Internet.
- ✓ **Seguridad y rentabilidad:** Puede programarse para que no se edite, copie o imprima. Además, con el software adecuado, puede codificarse para que sólo pueda consultarse previo pago de derechos.
- ✓ **Oportunidad y rapidez:** En la difusión y distribución de la información.
- ✓ **Inclusión de información tridimensional:** Puede incluirse sonido y video, así como la facilidad de su manipulación.

### **Desventajas de las publicaciones electrónicas** (Aguiar, 2003).

- ✓ Es indispensable un equipo para su lectura.
- ✓ No existe una reglamentación, se carece de un código de ética establecido.
- ✓ Favorece la proliferación de la información, además de que esta puede ser modificada por los usuarios.
- ✓ Se dificulta la identificación entre la información primaria y la secundaria.

### **1.2.2 Soluciones existentes a nivel internacional y nacional**

Muchos países en América Latina y Europa cuentan con soluciones para la edición y publicación electrónica de la GO, los cuales han aportado grandes éxitos en el desarrollo de las tecnologías de la información y las comunicaciones (TIC's) y unido a esto a la economía de dichos países.

Más relacionado con el tema de la legislación, se ha publicado en versión digital, una gran cantidad de números de la GO de la República de Cuba en sus diferentes ediciones (ordinaria, ordinaria especial, extraordinaria, extraordinaria especial y edición especial) desde el año 1991. Esta constituye una forma relativamente nueva de acceder a este tipo de documentos en nuestro país, la misma sirve de alternativa a la tradicional, por así decirlo, que consiste en consultar la legislación en su formato físico. Ambas presentan ventajas y desventajas pero indiscutiblemente la versión digital de la gaceta representa el futuro y el desarrollo.

Se realizó un estudio de soluciones que pudieran utilizarse como referencia para desarrollar el Sistema Web de la GO de la República de Cuba y se obtuvieron los siguientes resultados:

**Gaceta Oficial del Estado Plurinacional de Bolivia:** Tiene como objetivo realizar la publicación periódica, cronológica y oportuna de la GO de Bolivia, que incluya: Leyes, Decretos Supremos, Resoluciones Supremas, Tratados y Convenios Internacionales, Sentencias de Inconstitucionalidad, Boletines de Propiedad Industrial e Intelectual, Resoluciones Administrativas y otros.

**Boletín Oficial del Estado:** Este boletín es publicado por el estado español para dar a conocer las normas, actos y disposiciones legales que el ordenamiento jurídico considere que deben ser publicados. Brinda funcionalidades como la búsqueda avanzada de documentos, la visualización de los últimos anuncios que hayan salido, el acceso a otros boletines oficiales de otros sectores del país.

**Gaceta Oficial de la República Bolivariana de Venezuela:** Tiene como objetivo publicar las resoluciones y normas que dicta el estado Venezolano, entre sus funcionalidades se encuentran la navegación por el boletín donde se puede visualizar la publicación de números ordinarios y extraordinarios.

**Boletín Oficial de la República Argentina:** Brinda el acceso a los links de los boletines de países de América Latina.

**Gaceta Oficial de la República de Cuba:** El sitio actual de la Gaceta Oficial de Cuba tiene como objetivos: La publicación de disposiciones normativas, instrumentar nuevos recursos de información y diversificar las opciones de los destinatarios de la información publicada por la GO, para acceder a ella, conservar en soporte digital el tracto histórico de las normas jurídicas publicadas por la Gaceta, con vistas a hacerlo más eficiente, efectivo y potenciar las prestaciones de su archivo central.

Actualmente existe en nuestro territorio un sitio web para la publicación de la GO de Cuba, pero este no es capaz de satisfacer la problemática planteada en la presente investigación pues solo realiza la divulgación de la GO y los procesos de edición y publicación se continúan realizando de forma manual.

Basado en este diagnóstico el Centro de Gobierno Electrónico (CEGEL) de la Universidad de Ciencias Informáticas (UCI) de Cuba tomó la audaz decisión de realizar un sistema web que informatice los procesos de edición y publicación electrónica de la GO de Cuba.

### **1.3 Metodología de desarrollo de Software**

Uno de los temas más conocidos en el mundo de la informática es el de las metodologías de desarrollo de software, que tiene como objetivo aumentar la calidad del software que se produce en cada una de las fases de desarrollo, por medio de una mayor transparencia y control sobre el proceso; producir lo esperado en el tiempo y costo deseado (Penadés, 2003).

Todo sistema a implementar se rige por una metodología, que no es más que un conjunto de pasos a seguir en post del resultado final. Las metodologías de desarrollo de software están divididas en dos grupos que se conocen como, metodologías tradicionales y metodologías ágiles, siendo las tradicionales pensadas para hacer un uso exhaustivo de documentación durante todo el ciclo del proyecto y recomendada para los proyectos con grandes equipos de desarrollo, mientras que las ágiles hacen énfasis en la capacidad de dar respuesta a los cambios, promoviendo el trabajo en equipo y manteniendo una buena relación con el cliente.

A continuación la Tabla 1 representa las diferencias entre las metodologías tradicionales y las metodologías ágiles (Penadés, 2003).

Metodología Ágil	Metodología Tradicional
Basadas en heurísticas provenientes de prácticas de producción de código.	Basadas en normas provenientes de estándares seguidos por el entorno de desarrollo.
Especialmente preparados para cambios durante el proyecto.	Cierta resistencia a los cambios.
Impuesta internamente (por el equipo).	Impuestas externamente.
Proceso menos controlado, con pocos principios.	Proceso mucho más controlado, con numerosas políticas/normas.
No existe un contrato tradicional o al menos es bastante flexible.	Existe un contrato prefijado.
El cliente es parte de equipo de desarrollo.	El cliente interactúa con el equipo de desarrollo mediante reuniones.
Grupos pequeños (menos de 10 integrantes) y trabajando en el mismo sitio.	Grupos grandes y posiblemente distribuidos.
Pocos artefactos.	Más artefactos.
Pocos roles.	Más roles.
Menos énfasis en la arquitectura de software.	La arquitectura de software es esencial y se expresa mediante modelos.

**Tabla 1: Diferencia entre metodologías ágiles y no ágiles.**

Como resultado del análisis de las diferencias entre las metodologías ágiles y tradicionales, se llega a la conclusión de que, se hará uso de una metodología ágil para el desarrollo de la solución, determinando como elementos importantes para su selección los siguientes:

- El equipo de desarrollo está compuesto por 2 integrantes que trabajan en el mismo sitio y el proyecto es de corta duración.
- Se necesitan pocos roles pues el equipo de desarrollo es pequeño.
- La arquitectura del producto se va definiendo y mejorando a lo largo del proyecto.
- No existe un contrato firmado con un cliente.
- Se esperan cambios durante el desarrollo del proyecto en los requisitos.
- El cliente es parte del equipo de desarrollo.

Existen varias metodologías ágiles, de las más utilizadas a nivel mundial se realiza una comparación teniendo: vista del sistema como algo cambiante, colaboración entre los miembros del equipo y características propias de la metodología tales como: simplicidad, excelencia técnica, adaptabilidad y prácticas de colaboración. La Tabla 2 representa una comparación, donde los resultados y valores más altos representan una mayor agilidad (Highsmith, 2002).

	ASD	Crystal	DSDM	FDD	LD	Scrum	XP
Sistema como algo cambiante	5	4	3	3	4	5	5
Colaboración	5	5	4	4	4	5	5
Adaptabilidad	5	5	3	3	4	4	3
Simplicidad	4	4	3	5	3	5	5
Excelencia técnicas	3	3	4	4	4	3	4
Prácticas de colaboración	5	5	4	3	3	4	5

**Tabla 2: Metodologías ágiles más utilizadas.**

Después de realizar un estudio de los resultados de las diferentes metodologías existentes, se decide optar por la metodología Extreme Programming (XP) teniendo en cuenta los siguientes aspectos:

- XP se centra más en la programación o creación del producto que en la administración del proyecto.
- XP recomienda que las tareas a desarrollar sean desarrolladas por 2 personas en vez de una, como en otras metodologías ágiles logrando una mayor calidad del código al ser revisado y discutido mientras se escribe.
- Además de ser dentro de las metodologías ágiles, una de las más usadas en la UCI por lo que se posee más documentación sobre ella y existen varios especialistas que conocen sobre la misma.

### 1.3.1 Extreme Programming (XP)

Es una de las metodologías de desarrollo de software, enmarcada dentro de las metodologías ágiles de desarrollo. Constituye la metodología más utilizada dentro del grupo de las ágiles. Mientras que Rational Unified Process (RUP) intenta reducir la complejidad del software por

medio de estructura y la preparación de las tareas pendientes en función de los objetivos de la fase y actividad actual, XP, como toda metodología ágil, lo intenta por medio de un trabajo orientado directamente al objetivo, basado en una serie de valores, principios y prácticas aumentando la productividad a la hora de gestionar un software (Priolo, y otros, 2013).

### **Valores que promueve**

En el ciclo de vida de un proyecto los cambios van a aparecer y a veces el equipo de desarrollo no está preparado para enfrentarlos, XP desarrolla los siguientes valores para garantizar el éxito de un proyecto de desarrollo de software (Priolo, y otros, 2013):

- Comunicación: Crear software requiere de sistemas comunicados.
- Simplicidad: Empezar con lo necesario y requerido y trabajar desde ahí.
- Retroalimentación: Del sistema, del cliente, y del equipo.
- Valentía: Programa para hoy y no para mañana.
- Respeto: El equipo debe trabajar como uno, sin hacer decisiones repentinas.

### **Ciclo de vida**

El ciclo de trabajo de esta metodología consta de 4 fases: (Carillo, y otros, 2008).

**Planificación:** Es la fase donde se crean las Historias de Usuario, el Plan de iteración y el Plan de entrega.

**Diseño:** En esta fase se crean las tarjetas Cargos o Clases, Responsabilidad y Colaboración (CRC) las cuales definen una clase expresando las funcionalidades de esta y las otras clases con las que colabora.

**Desarrollo:** Es donde se definen las tareas de programación para que los desarrolladores tengan una guía para implementar todas las Historias de Usuario.

**Pruebas:** En esta fase se realizan las pruebas de aceptación a cada una de las Historias de Usuario para probar que cumplen con las funcionalidades que desea el cliente.

La metodología XP tiene principalmente dos objetivos, lograr una satisfacción con el cliente tratando de darle un software que verdaderamente necesita y en el momento que lo necesita y potenciar al máximo el trabajo en grupo identificando a cada miembro del equipo de desarrollo con el producto entregable al cliente.

### **1.4 Sistema de Gestión de Contenido (CMS)**

Un sistema gestor de contenidos (Content Manager System, por sus siglas en inglés CMS), tiene como función separar la gestión del contenido de su diseño. Brindan una mejor calidad a las páginas, facilitan la producción, administración, actualización de manera simple y dinámica y el mantenimiento de los sitios web (Sarduy, 2006).

Por lo tanto, un CMS tiene dos funciones principales: facilitar la creación de contenidos y la presentación de esos contenidos. Un CMS provee las herramientas necesarias para gestionar el ciclo de vida de los contenidos: creación, gestión, presentación, mantenimiento y actualización (Guervós, 2005).

Una de las características de los sistemas de gestión de contenidos, que además constituye una ventaja con respecto a su uso, es el hecho de posibilitar la inmediatez de la **publicación y edición de los contenidos**. Se obtiene entonces la agilidad necesaria para la divulgación de los contenidos en una institución (bilib.es, 2012).

A continuación se analizarán algunos de los CMS más utilizados para determinar cuál usar en la solución.

#### **1.4.1 WordPress**

WordPress es un CMS orientado a blogs o bitácoras en línea. Permite una estructura de soporte para que los usuarios de Internet puedan crear y administrar contenidos. Con WordPress los usuarios pueden cambiar, cuantas veces deseen, el diseño de su blog sin necesidad de cambiar el formato.

Su licencia GNU/GPL, su facilidad de uso, su enfoque hacia la elegancia y la estética, y otras características; convierten a WordPress en una de las plataformas de publicación personal más populares del mundo. Dentro de sus características principales se encuentra: la fácil instalación, actualización, personalización y uso en general. Es uno de los más populares sistemas de gestión de contenidos enfocado a la creación de weblogs. Desarrollado en Hypertext Preprocessor (PHP) y My Structured Query Language (MySQL), bajo licencia GPL (Sabin-Wilson, 2010).

#### **1.4.2 Drupal**

CMS muy conocido por la calidad de su código y por la seguridad que brinda, es estable y de actualización continua. Permite publicar artículos, imágenes, u otros archivos y servicios

añadidos como foros, encuestas, votaciones, blogs y administración de usuarios y permisos (Luis Alarcón, 2007).

Drupal está creado sobre estándares y tecnologías de código abierto, con licencia GPL y escrito en el lenguaje de programación PHP. Es desarrollado y mantenido por una activa comunidad de usuarios. Se destaca por la calidad de su código y de las páginas generadas, el respeto a los estándares de la web y un énfasis especial en la usabilidad y consistencia de todo el sistema.

El diseño de Drupal es idóneo para construir y gestionar comunidades en Internet. Si bien las características básicas de Drupal son suficientes como para lanzar un sitio completo, incluye: administración de usuarios, vocabulario, comentarios y sindicación, pero su potencialidad reside en su estructura modular que permite ser extendida y configurada por el administrador para distintas funciones por medio de módulos de fácil instalación (drupal.org.es, 2012).

Algunas de sus ventajas y características generales son:

- ✓ **Ayuda on-line:** Un robusto sistema de ayuda online y páginas de ayuda para los módulos del núcleo, tanto para usuarios como para administradores.
- ✓ **Búsqueda:** Todo el contenido en Drupal es totalmente indexado en tiempo real y se puede consultar en cualquier momento.
- ✓ **Código abierto:** El código fuente de Drupal está libremente disponible bajo los términos de la licencia GNU/GPL. Al contrario que otros sistemas de blogs o de gestión de contenido propietarios, es posible extender o adaptar Drupal según las necesidades.
- ✓ **Módulos:** La comunidad de Drupal ha desarrollado módulos que proporcionan infinidad de funcionalidades.
- ✓ **Personalización:** Tanto el contenido como la presentación pueden ser individualizados de acuerdo con las preferencias del usuario.
- ✓ **Autenticación de usuarios:** Los usuarios se pueden registrar e iniciar sesión de forma local o utilizando un sistema de autenticación externo. Para su uso en una intranet, Drupal se puede integrar con un servidor LDAP.
- ✓ **Permisos basados en roles:** Los administradores de Drupal no tienen que establecer permisos para cada usuario. En lugar de eso, pueden asignar permisos a un rol y agrupar los usuarios por roles.

### **1.4.3 Joomla**

El CMS Joomla es un sistema de código abierto construido con PHP, bajo una licencia GPL. Utiliza como base de datos MySQL. Es fácil de instalar, simple de manipular, fiable y disponible libremente para todo el que desee usarlo. La documentación es exhaustiva y concisa, interfaz de la administración muy intuitiva y poderosa. Entre sus características se encuentra: la versatilidad del sistema a través de plantillas, extensiones y adaptaciones; la posibilidad de hacer caché de páginas para mejorar el rendimiento; versiones imprimibles de páginas; indexado web; entre otras (joomla.org, 2013).

Joomla está dirigido a proyectos de pequeña y mediana envergadura que requieren Internet, como los sitios web corporativos, comunidades de usuario, tiendas online, etc. Para extraer todo el potencial de Joomla se necesita cierto grado de experiencia y conocimiento; pues su máxima versatilidad se obtiene de la integración, adaptación y desarrollo de nuevos módulos (Centro de Apoyo Tecnológico a Emprendedores, 2012).

### **1.4.4 Selección del sistema de gestión de contenido a utilizar**

Después del análisis de los sistemas de gestión de contenidos antes mencionados, se considera no utilizar el CMS Joomla porque posee una pobre gestión de usuarios, siendo este elemento necesario para la propuesta de solución. Se decide seleccionar para desarrollar la propuesta de solución el CMS Drupal en su versión 7.26, es importante destacar que Drupal ofrece todas las prestaciones y funcionalidades necesarias para gestionar los contenidos de un portal web. Proporciona gestión de usuarios, clasificación y edición de los contenidos y posee un potente sistema de seguridad basado en roles, que permite controlar a los diferentes módulos según los permisos que el administrador le haya definido a cada rol. También permite la combinación de módulos, que pueden ser modificados según las necesidades de los desarrolladores que lo utilizan, siendo una excelente opción para el trabajo que se necesita desarrollar en la presente investigación.

A continuación se describen un conjunto de herramientas y lenguajes que se requieren para el proceso de construcción de la propuesta de solución tomando como requerimiento la selección del CMS Drupal.

## **1.5 Sistema Gestor de Base de Datos (SGBD)**

Una base de datos es un conjunto de datos persistentes que refleja las interrelaciones y restricciones existentes en el mundo real, almacenados para su posterior uso. Es la

representación integrada de los conjuntos estructurados de entidades y sus interrelaciones. Esta representación informática (o conjunto estructurado de datos) debe poder ser utilizada de forma compartida por muchos usuarios de distintos tipos (Camps, 2005).

Un Sistema Gestor de Bases de Datos (SGBD) es un software dedicado a servir de interfaz entre la base de datos, el usuario y las aplicaciones que la utilizan. Tiene como propósito general crear y mantener una base de datos, asegurando su integridad, confidencialidad y seguridad (Ramakrishnan, 2000).

Los SGBD tienen como objetivos fundamentales:

- ✓ Independencia de los datos y los programas de aplicación
- ✓ Minimización de la redundancia
- ✓ Integración y sincronización de las bases de datos
- ✓ Integridad de los datos
- ✓ Seguridad y protección de los datos
- ✓ Facilidad de manipulación de la información
- ✓ Control centralizado

El CMS Drupal incorpora una capa de abstracción de base de datos que actualmente está implementada y mantenida para los SGBD MySQL, SQLite y PostgreSQL, a continuación se mencionan sus principales características.

### **1.5.1 PostgreSQL**

PostgreSQL permite crear nuevos tipos de datos y hacer herencias entre objetos. Posee transacciones, integridad referencial, vistas, y multitud de funcionalidades. Está considerado como el gestor de base de datos de código abierto más avanzado del mundo (Martínez, 2010).

PostgreSQL aproxima los datos a un modelo objeto-relacional y es capaz de manejar complejas reglas. Ejemplos de su avanzada funcionalidad son consultas SQL declarativas, optimización de consultas, soporte multi-usuario, transacciones, optimización de consultas, herencia, y arreglos (postgresql.org).

**Además ofrece múltiples ventajas como:**

- ✓ El código fuente está disponible para todos sin costo, haciéndolo extensible.

- ✓ Usa una estrategia de almacenamiento de filas llamada MVCC (siglas en inglés multiversion concurrency control) para conseguir una mejor respuesta en ambientes de grandes volúmenes.
- ✓ Está disponible en casi cualquier sistema Unix.
- ✓ Soporta integridad referencial, utilizada para garantizar la validez de los datos de la base de datos.
- ✓ Posee una amplia variedad de tipos nativos.
- ✓ Tiene mejor soporte para triggers y procedimientos en el servidor.
- ✓ Altamente adaptable a las necesidades del cliente.
- ✓ Extensiones para alta disponibilidad, nuevos tipos de índices, datos espaciales y minería de datos.
- ✓ Probado con torres de discos duros de alta disponibilidad.

### **1.5.2 MySQL**

MySQL es un sistema gestor de base de datos relacional, con licencia dual, conocido y ampliamente usado por su simplicidad y notable rendimiento. Opera en una arquitectura cliente/servidor. Permite la fácil conectividad, gran velocidad de solicitudes y seguridad por lo que se utiliza para acceder a bases de datos desde Internet (mysql.com, 2012).

Las principales características de este gestor de bases de datos son:

- ✓ Soporta gran cantidad de tipos de datos para las columnas.
- ✓ Gran portabilidad entre sistemas.
- ✓ Soporta hasta 32 índices por tabla.
- ✓ Gestión de usuarios y contraseña, manteniendo un buen nivel de seguridad en los datos.
- ✓ Alta velocidad a la hora de realizar las operaciones.
- ✓ Su bajo consumo, lo hacen apto para ser ejecutado en una máquina con escasos recursos sin ningún problema.
- ✓ Gran facilidad de configuración e instalación.
- ✓ Gran rapidez, facilidad de uso.

El SGBD MySQL tiene como desventaja que es un producto de licencia dual, se ofrece bajo una licencia privativa, y además bajo la licencia de GNU/GPL. Esto implica que aunque es desarrollado por una comunidad pública, el copyright del código está en poder del autor individual. MySQL es propietario y está patrocinado por una empresa privada, que posee el copyright de la mayor parte del código. Además en la actualidad la versión de código abierto no se le proporciona soporte.

### **1.5.3 SQLite**

SQLite es un SGBD relacional. Constituye un proyecto de dominio público y se encuentra contenido en una pequeña biblioteca. A diferencia de los SGBD cliente-servidor, el motor de SQLite no es un proceso independiente con el que el programa principal se comunica. En lugar de eso, la biblioteca se enlaza con el programa pasando a ser parte integral del mismo. El programa utiliza la funcionalidad de SQLite a través de llamadas simples a subrutinas y funciones. Esto reduce la latencia en el acceso a la base de datos, debido a que las llamadas a funciones son más eficientes que la comunicación entre procesos. En su versión 3, SQLite permite bases de datos de hasta 2 Terabytes de tamaño (Hipp, 2013).

### **1.5.4 Selección del sistema de gestión de base de datos a utilizar**

Luego de realizar el estudio de los sistemas gestores de base de datos PostgreSQL, SQLite y MySQL, se considera que los tres podrían ser usados en el desarrollo de la propuesta de solución. Se decide emplear como herramienta para almacenar y gestionar los datos el SGBD PostgreSQL en su versión 9.1, por las experiencias adquiridas en otros proyectos. PostgreSQL brinda la posibilidad de almacenar grandes volúmenes de datos sin perder la estabilidad y la disponibilidad.

## **1.6 Tecnología y herramientas utilizadas en la implementación del sistema web**

A continuación se describen las principales características de las tecnologías asociadas al CMS Drupal, conjuntamente con las herramientas que apoyarán y facilitarán el desarrollo de la propuesta de solución.

### **1.6.1 Lenguaje de Modelado**

Un lenguaje de modelado es un conjunto estándar de símbolos y formas que facilita la modelación de un diseño de software. A continuación se muestran los lenguajes de modelados utilizados en la solución.

## Lenguaje de Modelado Unificado (UML)

El Lenguaje Unificado de Modelado (UML, por sus siglas en inglés, Unified Modeling Language) es un lenguaje gráfico para visualizar, especificar y documentar cada una de las partes que comprende el desarrollo de software. UML permite una forma de modelar conceptualmente procesos de negocio y funciones de sistema, además de elementos concretos como son esquemas de base de datos, componentes de software reutilizables y escribir clases en un lenguaje determinado (Flores, y otros, 2013).

UML proporciona mecanismos de extensibilidad, los cuales permiten a sus usuarios refinar su sintaxis y su semántica, por lo tanto, puede ajustarse a un sistema, proyecto, proceso de desarrollo específico si es necesario. Además cuenta con un vocabulario que incluye tres categorías (Larman, 1999):

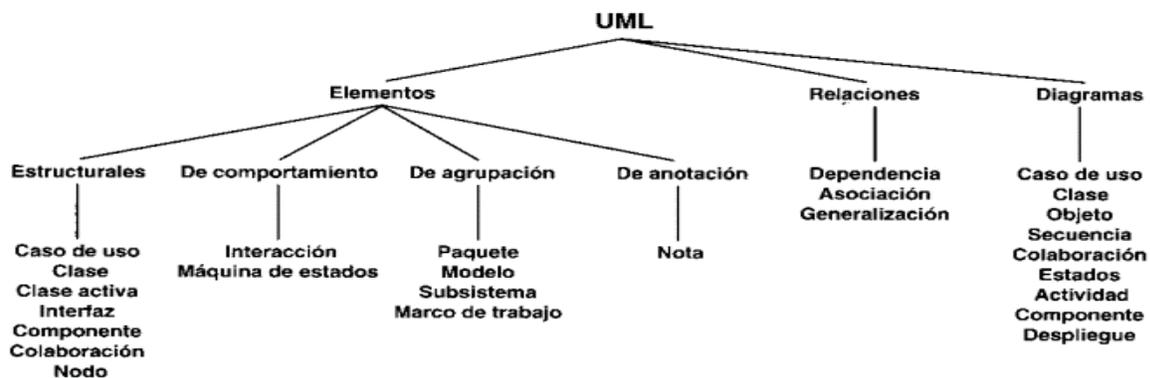


Figura 2: Vocabulario de UML (Jacobson, y otros, 2000).

## Business Process Management Notation (BPMN)

BPMN proporciona a los negocios la capacidad de entender sus procedimientos internos en una notación gráfica, facilitando a las organizaciones la habilidad para comunicar esos procedimientos de una manera estándar (Hernández, 2010).

Sus principales objetivos son (Hernández, 2010):

- ▶ Proveer una notación que sea fácilmente entendida por todos los usuarios.
- ▶ Asegurar que los lenguajes para la ejecución de los procesos de negocio puedan ser visualizados con una notación común.
- ▶ BPMN es usado para comunicar una amplia variedad de información a una amplia variedad de audiencias.

### **1.6.2 Lenguaje del lado del cliente**

Los lenguajes del lado del cliente son los asimilados directamente por el navegador y no necesitan pre-tratamiento.

#### **Hyper Text Markup Language (HTML)**

Hyper Text Markup Language (HTML o Lenguaje de Marcas de Hipertexto) es un lenguaje de marcado para construir páginas web, que se utiliza normalmente en la World Wide Web. Básicamente es un conjunto de etiquetas de marcado para definir la estructura y los contenidos que compondrán las páginas web. Permite combinar textos, imágenes, sonidos, vídeos y enlaces a otras páginas. Es un lenguaje sencillo y fácil de aprender, además posibilita combinar dentro de su código scripts otros lenguajes como PHP y JavaScript. Los documentos HTML son archivos de texto plano que pueden ser creados mediante cualquier editor de texto, no obstante existen programas concebidos específicamente para editar páginas web en HTML (w3.org, 2012).

#### **JavaScript**

JavaScript es el lenguaje interpretado orientado a objetos, desarrollado por Netscape. Se utiliza en millones de páginas web y aplicaciones de servidor en todo el mundo, para agregar funcionalidades, validar formularios, comunicarse con el servidor y otros elementos. Es fácil de aprender, puede ser usado por profesionales y quienes se inician en el desarrollo y diseño de sitios web. La sintaxis básica es similar a Java y C++, con la intención de reducir el número de nuevos conceptos necesarios para aprender el lenguaje. Aunque muchos confunden JavaScript con Java, ambos son lenguajes diferentes y tienen sus características singulares. JavaScript no requiere de compilación pues funciona del lado del cliente y son los navegadores los encargados de interpretar estos códigos. (developer.mozilla.org, 2011).

#### **Cascading Style Sheets (CSS)**

Cascading Style Sheets (CSS) conocido en español como Hojas de Estilo en Cascada. Es un mecanismo simple que permite controlar la apariencia de una página web. Describe cómo se va a mostrar un documento en la pantalla, cómo se va a imprimir o incluso cómo va a ser pronunciada la información presente en ese documento a través de un dispositivo de lectura. Esta forma de descripción de estilos le permite a los desarrolladores web controlar el estilo y formato de múltiples páginas web al mismo tiempo (w3c.es, 2008).

### **1.6.3 Lenguaje del lado del servidor**

Los lenguajes del lado del servidor son los lenguajes que se ejecutan por el propio servidor y son enviados al cliente en un formato claro.

#### **Hypertext Preprocessor (PHP)**

PHP es un acrónimo que significa en inglés Hypertext Preprocessor, es un lenguaje centrado en la programación de scripts del lado del servidor. Se caracteriza por su rapidez y simplicidad, usado para complementar el lenguaje HTML, puede ser incrustado en HTML, por lo que convierte una página web estática en dinámica. Contiene una gran librería de funciones y mucha documentación, especialmente adecuado para el desarrollo web. Además es de código abierto e independiente de plataforma. Puede ser utilizado en cualquiera de los principales sistemas operativos del mercado, y soporta la mayoría de servidores web de hoy en día, incluyendo Apache, IIS, y muchos otros. Una de sus características más potentes y destacables es capacidad de conexión con la mayoría de los manejadores de base de datos: MySQL, PostgreSQL, Oracle, MS SQL Server, entre otras (Olson, 2012).

### **1.6.4 Herramienta de Modelado**

Las herramientas CASE son un conjunto de programas y ayudas que dan asistencia a los analistas, ingenieros de software y desarrolladores, durante todos los pasos del Ciclo de Vida de desarrollo de un Software (inei.gob.pe).

#### **Visual Paradigm for UML**

Visual Paradigm for UML es una potente herramienta que permite visualizar y diseñar elementos de software y soporta el ciclo de vida completo del desarrollo de software. Una de sus ventajas sobre las demás herramientas CASE es condición multiplataforma, por lo que tiene la capacidad de ejecutarse sobre diferentes Sistemas Operativos. Es fácil de usar y presenta una agradable interfaz para interactuar con el usuario. Se integra con el Visio para importar imágenes del mismo para realizar los diagramas de despliegue y genera documentación para el proyecto en HTML, MS Word y PDF. Además, importa y exporta diagramas en XML y como imágenes (ya sea con extensiones jpg o png). Otra característica importante es la generación del código C#, VB.NET, Object Definition Language (ODL), Flash ActionScript, Delphi, Perl, y Ruby. La ingeniería inversa que proporciona también apoya clase Java, .NET, .DLL y JDBC (inei.gob.pe).

Se decide trabajar con la herramienta CASE Visual Paradigm pues brinda facilidades para el diseño de los diagramas, permite el diseño del producto de forma rápida y con calidad, así como su documentación, puede generar código y realizar ingeniería inversa a diferentes lenguajes de programación. Es también muy robusta, usable y portable. Además la Universidad cuenta con la licencia para su uso.

### **1.6.5 Entorno de Desarrollo Integrado (IDE)**

NetBeans IDE es un entorno de desarrollo de código abierto, una herramienta para que los programadores puedan escribir, compilar, depurar y ejecutar programas. Está escrito completamente en Java, pero se puede utilizar para cualquier otro lenguaje de programación. Dispone de soporte para crear interfaces gráficas de forma visual, desarrollo de aplicaciones web, control de versiones, colaboración entre varias personas y creación de aplicaciones compatibles con teléfonos móviles. Existe además un número importante de módulos para extender el NetBeans IDE. Puede ser instalado en todos los sistemas operativos que soportan Java, como Microsoft, Linux y Mac OS. El NetBeans es intuitivo y completo para PHP, pues contiene un editor para este lenguaje con completamiento de código y resaltado de código sintáctico y semántico, chequeo de errores, depurador de código PHP (netbeans.org, 2012).

### **1.6.6 Herramienta de prueba de software**

#### **Apache JMeter**

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

Se puede utilizar para probar el rendimiento tanto estático como dinámico de los recursos (ficheros, objetos de Java, bases de datos, servidores de FTP24 y otros). Ofrece la posibilidad de visualizar varias estadísticas de carga y la reutilización de las pruebas en el desarrollo de otras nuevas (Martínez, 2009).

### **1.6.7 Servidor Web**

#### **Apache Server**

La piedra angular de cualquier sistema web es con toda seguridad el servidor Web; el software encargado de atender las peticiones de los clientes y enviarles las páginas Web solicitadas. Existen multitud de paquetes software para montar servidores Web, muchos de ellos distribuidos como software libre. El más popular de todos es el Apache.

El Apache, fue hecho para proveer un alto grado de calidad y fortaleza para las implementaciones que utilizan el protocolo HTTP. Está ligado a la plataforma (Linux, Windows, UNIX) sobre la cual, los individuos o instituciones pueden construir sistemas confiables con fines experimentales, o para resolver un problema específico de la organización (Kabir, 2003).

Teniendo en cuenta las características potenciales del servidor web Apache 2.2.21, y rendimiento del mismo se selecciona, además de ser el Servidor HTTP más usado desde 1996 (70 % de los sitios Web en 2005) con Software de código abierto, que cuenta con una autenticación en el acceso a la base de datos y altamente configurable y extensible.

### **1.7 Conclusiones parciales**

En este capítulo la realización del estudio de los sistemas homólogos permitió determinar las principales características, funcionalidades y servicios que se encuentran presentes en los sistemas de gestión de información legal, y deben ser incluidos en la solución del problema. Además la descripción de los principales conceptos permitió adquirir una mayor comprensión de los temas relacionados con el objeto de estudio, y la necesidad de implementar un sistema web que posea una buena usabilidad y accesibilidad, así como una correcta gestión de los contenidos. Por otra parte el análisis sobre las potencialidades, ventajas y desventajas de las diferentes metodologías, herramientas y tecnologías permitió alcanzar los conocimientos necesarios para seleccionar las adecuadas en el desarrollo de la solución del problema quedando como plataforma de desarrollo el sistema gestor de contenidos Drupal, como gestor de bases de datos para el almacenamiento de la información: PostgreSQL y como servidor web para el hospedaje de la aplicación: Apache.

## **Capítulo 2: Planificación, diseño**

En el capítulo se presentarán las características y el funcionamiento del sistema, donde se tiene en cuenta el estilo arquitectónico y los patrones de diseño aplicables al CMS Drupal. Se realizará una descripción del flujo actual de los procesos que se desarrollan en el sistema web para la edición y publicación de la GO de la República de Cuba. Al mismo tiempo se mostrará a través de un modelo de dominio los principales objetos del sistema y se identificarán los requisitos funcionales y los no funcionales que deben estar presentes en la solución. Además, se describe el funcionamiento del CMS y los módulos de la ampliación del sistema.

### **2.1. Descripción del sistema**

La propuesta de solución de la presente investigación, consiste en un sistema para la edición y publicación electrónica de la GO de la república de Cuba. Su desarrollo se basa en la utilización de las últimas versiones estables de las herramientas definidas, lo que brinda un mejor soporte y una mayor flexibilidad y adaptabilidad de la solución. El sistema a desarrollar deberá dar cumplimiento a las necesidades de realizar un correcto proceso de edición y publicación de la GO de la República de Cuba por parte del MINJUS, con el objetivo de darle a conocer a todos los usuarios las leyes que se promulgan en nuestro país.

A través de este sistema, se podrá consultar la información concerniente a los procesos de edición y publicación de la GO, comprobar el cumplimiento de los plazos establecidos por ley, ofrecer una plataforma de trabajo colaborativo para las instituciones que intervienen en el proceso legislativo cubano, así como permitir el proceso de edición de la Gaceta y consecuentemente su publicidad digital en el sitio de la Gaceta Oficial, lo cual permite introducir mejoras al proceso de las ediciones anteriores, las ediciones del mes, documentos importantes relacionados a la Legislación Cubana, los símbolos de la nación, la interacción de los diferentes actores y gestionar la información generada por los procesos de edición y publicación de la GO de manera que contribuya al establecimiento de un canal de comunicación y servicios a la comunidad de usuarios de dichas entidades, para ello se debe ajustar a las funcionalidades siguientes:

- ✓ Configurar y Administrar el sistema.
- ✓ Gestionar la tramitación de documentos normativos para publicar en la Gaceta Oficial.
- ✓ Gestionar la conciliación a partir de los comentarios que se emiten por los organismos al Ministerio acerca de las nuevas leyes contenidas en la Gaceta.

- ✓ Controlar los términos establecidos para la realización de los procesos.
- ✓ Visualizar en formato PDF la edición de la GO y asignarle los metadatos para su publicación.

### Personas que interactúan con el Sistema:

- **Administrador:** Persona que tendrá todos los privilegios del sistema, será el encargado de gestionar el rol que tendrán los usuarios registrados en el sistema.
- **Editor:** Persona que tendrá el privilegio de gestionar los documentos legislativos para conformar correctamente la GO. Una vez autenticado en el sistema podrá realizar todas las operaciones que su nivel de privilegio le permita acceder.
- **Publicador:** Persona que tendrá el privilegio de gestionar cualquier documento que haya pasado anteriormente el proceso de edición, así también, consultar cualquier otro artículo. Una vez autenticado en el sistema podrá realizar todas las operaciones que su nivel de privilegio le permita acceder.
- **Revisor:** Persona que tendrá privilegios para revisar que el documento enviado por los emisores oficiales de disposiciones jurídicas sea legislativo y cumpla con los requisitos formales de la GO, antes de que este pase a la fase de edición.
- **Promulgador:** Persona que promueve la ley, puede ser de la Fiscalía General de la República (FGR), del Tribunal Supremo Provincial (TSP) o de los Organismos de la Administración Central del Estado (OACE).
- **Usuario invitado:** Usuario que navega en el sistema web sin estar autenticado.

### 2.2. Modelado del negocio

Los diagramas de procesos de negocio son una representación gráfica de los procesos de una organización, que muestran principalmente cómo y por quién es llevada a cabo las actividades que generan valor para la organización (Hernández, 2010). Muestran también:

- Los actores involucrados en los procesos.
- Cuáles son las actividades operativas distinguibles.
- Que actividades son ejecutables y por quien.
- Cuáles son las entradas y salidas de actividades.
- Cuál es la secuencia de las actividades.

- Los recursos consumidos y los eventos que dirigen el proceso.

Las cuatro categorías básicas de elementos que se pueden encontrar en un diagrama de proceso de negocio son (Hernández, 2010):

- Objetos de Flujo.
- Objetos de Conexión.
- Roles.
- Artefactos.

Se logró conformar el modelo de negocio el cual permitió describir los procesos de edición y publicación electrónica.

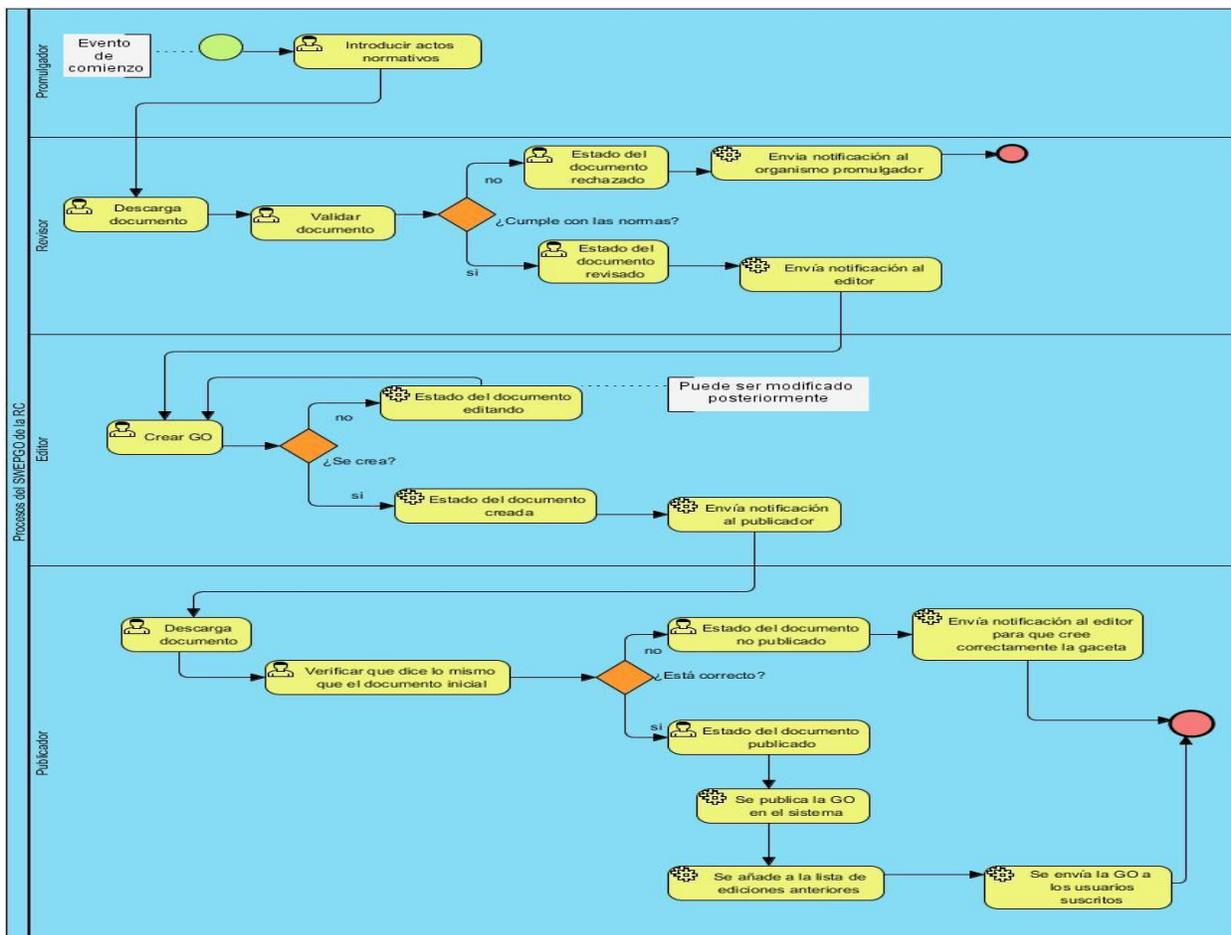


Figura 3: Diagrama de Proceso de Negocio.

### 2.3. Planificación

La metodología de desarrollo de software XP muestra la planificación como un permanente diálogo entre el usuario final y el equipo de desarrollo; definiéndose lo que el software tiene que resolver para que genere algún valor, la parte del software que debe hacerse primero, lo que se necesita hacer para saber si el negocio requiere o no de la informatización a través de un software, fechas de entrega de las primeras versiones del software, la organización del trabajo y el equipo de desarrollo y por último se detalla dentro de una versión del producto los problemas que deben de ser resueltos con carácter urgente (Noble, y otros, 2004).

#### 2.3.1 Historias de usuarios detectadas

En el entorno de XP las Historias de Usuario (HU) son utilizadas para describir las funcionalidades que serán añadidas al sistema, son asignadas al desarrollador encargado de la programación con un número de horas de desarrollo estimado, descritas por el cliente que forma parte del equipo de desarrollo. Estas historias de usuario están documentadas en el expediente de proyecto en la plantilla de HU. En el documento se presenta la especificación de una de ellas.

Durante la fase de planificación se identificaron 21 HU que a continuación se muestran, representando cada una las funcionalidades del sistema.

- Gestionar usuarios.
- Gestionar roles.
- Autenticar usuario.
- Subir al sistema documento legislativo.
- Realizar búsqueda de la GO.
- Listar Ediciones anteriores de la GO.
- Listar documento legislativo por estados.
- Gestionar la GO.
- Publicar la GO.
- Cambiar el estado al documento legislativo.
- Imprimir la GO desde el sistema.
- Descarga en formato PDF de la GO.

- Enviar notificación a los roles del cambio de estado.
- Suscribirse al sistema.
- Enviar por correo último número de la GO.
- Eliminar la suscripción del sistema.
- Visualizar en la web en formato PDF la GO.
- Gestionar Comentario.
- Mostrar detalles de una GO.
- Verificar fortaleza de contraseña de usuario.
- Cerrar sesión.

Se describe a continuación la HU: **Gestionar la GO.**

<b>Historia de Usuario</b>	
<b>Número:</b> 8	<b>Nombre de la Historia de Usuario:</b> Gestionar la GO.
<b>Cantidad de modificaciones a la Historia de Usuario:</b> Ninguna.	
<b>Usuario:</b> Ledier Hernández Martínez.	<b>Iteración asignada:</b> 1.
<b>Prioridad en negocio:</b> Alta.	<b>Puntos estimados:</b> 3 semanas.
<b>Riesgo en desarrollo:</b> Alta.	<b>Puntos reales:</b> 3 semanas.
<b>Programador responsable:</b> Ledier Hernández Martínez.	
<b>Descripción:</b> Consiste en crear una GO a través de una planilla que genera el sistema con los diferentes campos que tiene, así como modificar una que no se haya terminado y tenga estado de "Editando".	
<b>Observaciones:</b> En caso de que se termine una GO y tome estado de "Creada" y se quiera modificar de nuevo pues tiene que pasar por el proceso de edición y publicación desde el comienzo.	

**Tabla 3: Historia de Usuario: Gestionar GO.**

### **2.3.2 Requisitos del sistema**

La especificación de los requisitos del software se encarga de definir y describir de forma clara, consistente, compacta, y sin ambigüedades, el comportamiento del sistema. Permite estimar costos, tiempo y recursos. Se dividen en los siguientes grupos.

#### **Requisitos funcionales del sistema (RF)**

A continuación se listan los requisitos funcionales detectados para el sistema:

##### **HU: Autenticar usuario.**

RF1: Autenticar usuario.

##### **HU: Gestionar usuario.**

RF2: Modificar usuario.

RF3: Insertar usuario.

RF4: Eliminar usuario.

##### **HU: Gestionar roles.**

RF5: Modificar rol.

RF6: Adicionar rol.

RF7: Eliminar rol.

##### **HU: Verificar fortaleza de contraseña de usuario.**

RF8: Verificar fortaleza de contraseña de usuario.

##### **HU: Cerrar sesión.**

RF9: Cerrar sesión.

##### **HU: Subir al sistema documento legislativo.**

RF10: Subir al sistema documento legislativo.

##### **HU: Realizar búsqueda de la GO.**

RF11: Realizar búsqueda simple de la GO.

RF12: Realizar búsqueda avanzada de la GO.

##### **HU: Listar Ediciones anteriores de la GO.**

RF13: Listar Ediciones anteriores de la GO.

**HU: Listar documento legislativo por estados.**

RF14: Listar documento legislativo por estados.

**HU: Cambiar el estado al documento legislativo.**

RF15: Cambiar el estado al documento legislativo.

**HU: Gestionar la GO.**

RF16: Crear GO.

RF17: Modificar GO.

**HU: Enviar notificación a los roles del cambio de estado.**

RF18: Enviar notificación a los roles del cambio de estado.

**HU: Publicar la GO.**

RF19: Mostrar en la portada principal último número de la GO.

RF20: Adicionar la GO al sistema.

**HU: Imprimir la GO desde el sistema.**

RF21: Imprimir la GO desde el sistema.

**HU: Descarga en formato PDF de la GO.**

RF22: Descarga en formato PDF de la GO.

**HU: Visualizar en la web en formato PDF la GO.**

RF23: Visualizar en la web en formato PDF la GO.

**HU: Suscribirse al sistema.**

RF24: Suscribirse al sistema.

**HU: Enviar por correo último número de la GO.**

RF25: Enviar por correo último número de la GO.

**HU: Eliminar la suscripción del sistema.**

RF26: Eliminar la suscripción del sistema.

**HU: Gestionar Comentario.**

RF27: Enviar comentarios.

RF28: Aprobar comentarios.

RF29: Mostrar comentarios.

**HU: Mostrar detalles de una GO.**

RF 30: Mostrar detalles de una GO.

### **Requisitos no funcionales del sistema (RnF)**

Los RnF detallan las propiedades o cualidades que el producto debe tener, aumentan funcionalidad, pues hacen al producto fácil de usar, rápido y confiable. (Oliveros, 2012).

A continuación se muestran los RnF agrupado por las siguientes clasificaciones:

#### **Apariencia o Interfaz externa**

**RNF 1:** La interfaz gráfica de la aplicación debe concebirse con un ambiente amigable.

**RNF 2:** La interfaz contará con menús desplegables para agilizar y acelerar su utilización.

**RNF 3:** La interfaz de comunicación de la aplicación debe ser segura y confiable.

**RNF 4:** El sistema estará optimizado para una resolución de 1024x768 píxeles.

**RNF 5:** Identificación de colores y formatos con La República de Cuba.

#### **Usabilidad**

**RNF 6:** El sistema debe presentar una interfaz amigable que permita la fácil interacción con el mismo, llegar de manera rápida y efectiva a la información buscada.

**RNF 7:** Se proporcionará una interfaz de manejo cómoda, que posibilite una rápida adaptación, tanto para los usuarios con conocimientos avanzados en informática como los usuarios más inexpertos.

**RNF 8:** Debe tener buena visibilidad en los principales navegadores web.

#### **Eficiencia**

**RNF 9:** El sistema debe ser capaz de responder con rapidez a las peticiones de los usuarios, demorando como promedio en una transición, de uno (1) a tres (3) segundos aproximadamente.

**RNF 10:** Se esperan alrededor de cien usuarios concurrentes.

### **Fiabilidad**

**RNF 11:** El sistema debe estar disponible el 98% del tiempo, previendo el uso del 2% del tiempo para soporte, corrección de errores y actualizaciones.

**RNF 12:** El tiempo medio de corrección de errores no debe exceder las 72 horas.

**RNF 13:** Debe registrar cualquier fallo de su funcionamiento para poder ser solucionado por los administradores.

### **Seguridad**

**RNF 14:** El acceso será controlado con nombres de usuario y contraseñas.

**RNF 15:** Se utilizará métodos de encriptación para cumplir con la autenticación segura (SSL).

**RNF 16:** La información manejada por el sistema será objeto de cuidadosa protección contra la corrupción y estados inconsistentes.

**RNF 17:** Identificar al usuario antes de que pueda realizar cualquier acción sobre el contenido del portal. Garantizar que cada usuario realice las actividades permitidas a su rol.

**RNF 18:** Garantizar que la información sea editada únicamente por quien está autorizado.

**RNF 19:** Garantizar que las funcionalidades del sistema se muestren de acuerdo al nivel de usuario que esté activo.

**RNF 20:** Protección contra acciones no autorizadas o que puedan afectar la integridad de los datos.

**RNF 21:** Verificación sobre acciones irreversibles (eliminaciones).

### **Soporte**

**RNF 22:** Servidor web Apache 2.2.21.

**RNF 23:** Por parte del cliente se requiere un navegador capaz de interpretar HTML 5, JavaScript 1.6 y CSS 3.

**RNF 24:** Gestor de base de datos PostgreSQL en su versión 9.1 o superior.

**RNF 25:** Se usará como lenguaje de programación: PHP 9.1 y como sistema de gestión de contenidos Drupal 7.26.

**RNF 26:** Debe dar la posibilidad de ser mejorado y de incorporarle nuevos servicios en caso de necesitarlos.

### Hardware

**RNF 27:** Los servidores proxy, web y de base de datos deben poseer como mínimo 2 GB de memoria RAM.

### Requisitos de licencia

**RNF 28:** Uso de la licencia GNU/GPL versión 2 para el CMS Drupal.

**RNF 29:** Uso de la licencia BSD de PostgreSQL.

**RNF 30:** Uso de la licencia Apache Software.

**RNF 31:** Uso de la licencia PHP License.

Todas consideradas licencias libres y flexibles aprobadas por la Fundación de Software Libre.

### 2.3.3 Técnicas de validación de requisitos

Con el objetivo de ratificar que los requisitos del software obtenidos definen el sistema que el cliente desea se llevó a cabo un proceso de validación de los mismos, para el cual se emplearon las siguientes técnicas:

**Revisiones de los requisitos:** Se realizaron revisiones a los requisitos por parte del equipo de desarrollo y por el cliente. Las revisiones por parte del equipo de desarrollo no generaron no conformidades, en cambio con el cliente se realizaron 3 revisiones. En la primera se agregaron 3 RF, en la segunda el cliente quedó satisfecho con el trabajo realizado por el equipo de desarrollo en cuanto a la primera revisión, sin embargo le realizó modificaciones a 1 RF para su perfeccionamiento y en la tercera revisión el cliente quedó satisfecho completamente con el trabajo realizado por el equipo de desarrollo y se aprobaron los requisitos.

**Generación de casos de prueba:** Como parte del proceso de validación de los requisitos funcionales del sistema fueron diseñados casos de pruebas para cada HU, en el Anexo 2 se muestra los casos de pruebas de las funcionalidades del sistema.

### 2.3.4 Métricas usadas para la validación de la calidad de los requisitos

La validación de requisitos tiene como objetivo demostrar que la definición de los requisitos define realmente el sistema que el usuario necesita o el cliente desea. La validación se realiza una vez definidos los requisitos del sistema para asegurar que el análisis de los mismos y los resultados obtenidos durante la definición de requisitos son correctos. De esta manera se asegura que todos los requisitos del sistema han sido establecidos sin ambigüedad, sin

inconsistencias, sin omisiones, que los errores detectados hayan sido corregidos. (Sánchez y otros, 2010).

Para medir la estabilidad de los requisitos, se aplicó la técnica “ **Estabilidad de requisitos** ” pues es necesario lograr una estabilidad en los requisitos para el correcto funcionamiento de los demás flujos de trabajo. El objetivo de esta métrica es medir la estabilidad de los requerimientos para asegurar su adecuación antes de pasar al próximo flujo de trabajo.

Se identificaron un total de 30 requisitos funcionales, de los cuales 3 requisitos fueron sometidos a modificaciones (estos 3 por actualización), por tanto:

$$ETR = \left[ \frac{RT - RM}{RT} \right] * 100$$

$$ETR = [(30 - 3) / 30] * 100 = 90.$$

Dónde:

RT: Total de requisitos definidos.

RM: Cantidad de requisitos modificados.

Al medir la estabilidad de los requisitos se llegó a la conclusión que los requisitos definidos son estables, por lo tanto, es confiable trabajar el análisis y diseño sobre ellos.

Para cuantificar la especificidad de los requisitos se aplicó la técnica “**Especificidad de requisitos**” de la siguiente manera:

$$Q_1 = \frac{n_{ui}}{n_r}$$

$$Q_1 = 28/30 = 0.93$$

Dónde:

Q1: Especificidad de los requisitos.

nr: Total de requisitos definidos.

nui: Total de requisitos para los que los revisores tuvieron interpretaciones idénticas.

Esta característica asegura mayor calidad en el proceso de especificación.

Para medir que los requisitos identificados con el cliente fueran correctos se aplicó la técnica “**Grado de validación**”, que se realiza en consenso del equipo de desarrollo al contrastar lo que desea el cliente con la posibilidad real de implementarlo. El grado de validación de los requerimientos mide la corrección en la definición de los requerimientos:

$$Q_3 = \frac{n_c}{(n_c * n_{nv})} \quad f$$

$$Q_3 = 30/(30+0) = 30/30 = 1$$

Dónde:

Q3: grado de validación de los requerimientos.

nc: número de requerimientos que se han validado como correctos.

n nv: número de requisitos no validados aún.

De acuerdo al resultado obtenido como el valor óptimo es 1 se concluye que existe un alto nivel de corrección en la definición de los requerimientos.

### 2.3.5 Estimación de esfuerzo por historias de usuarios

Los puntos de estimación de esfuerzo por HU es el tiempo de implementación para cada una de estas, cada punto representa una semana ideal de trabajo, por lo que si se acuerda en dar un punto a una HU determinada, esto significaría que solo se necesita una semana de trabajo para esta HU, y así sucesivamente. Para establecer la complejidad de las HU se tuvo en cuenta una series de indicadores que evalúan a cada una de ellas, dando como resultado si la complejidad es alta, media o baja. Estos indicadores están documentados en el expediente de proyecto en la plantilla: Complejidad de las HU.

Historia de Usuario	Puntos de Estimación
Autenticar usuario. Gestionar usuario. Gestionar roles. Verificar fortaleza de contraseña de usuario. Cerrar sesión.	2
Subir al sistema documento legislativo. Cambiar el estado al documento legislativo.	1

Gestionar la GO. Enviar notificación a los roles del cambio de estado.	3
Publicar la GO. Imprimir la GO desde el sistema. Descarga en formato PDF de la GO. Visualizar en la web en formato PDF la GO.	3
Listar Ediciones anteriores de la GO. Listar documento legislativo por estados.	1
Subscribirse al sistema. Enviar por correo último número de la GO. Eliminar la suscripción del sistema.	1
Mostrar detalles de una gaceta oficial. Realizar búsqueda de la GO. Gestionar Comentario.	2

**Tabla 4: Tabla de estimación de esfuerzo por HU.**

### 2.3.6 Plan de iteraciones

Como parte del ciclo de vida de un proyecto usando la metodología XP se crea el plan de duración de cada una de las iteraciones que se han definido, que tiene como objetivo mostrar la duración y el orden en que serán implementadas las HU dentro de cada iteración. Para la solución se han definido 21 HU divididas en 2 iteraciones, de acuerdo a los intereses del cliente, para una duración total del proyecto de 13 semanas como se muestra.

**Iteración 1:** En esta iteración se implementan las HU con prioridad **alta**. Al culminar el ciclo de esta primera iteración se podrá contar con las funcionalidades descritas en las HU: 1, 2, 3, 5, 8, 9, 17. Además se tendrá la primera versión de prueba que se le mostrará al cliente con el objetivo de hacer cambios en dependencia a la aceptación del mismo.

**Iteración 2:** En esta iteración se implementan las HU con prioridad **media y baja**. Con la culminación de esta se tendrán implementadas las peticiones del cliente descritas en las HU: 4, 6, 7, 10, 11, 12, 13, 14, 15, 16, 18, 19, 20, 21 obteniendo como resultado la primera versión del producto final.

### 2.3.7 Plan de duración de iteraciones

Siguiendo el desarrollo de la Metodología XP se crea el plan de duración de cada una de las iteraciones, el cual tiene como objetivo reflejar la duración de cada iteración y el orden en que serán implementadas las HU en estas, para dar una idea aproximada de la confección del sistema en su totalidad.

Iteración	Orden de la HU a implementar	Duración
<b>Iteración # 1</b>	HU: Autenticar usuario. HU: Gestionar usuario. HU: Gestionar roles. HU: Realizar búsqueda de la GO. HU: Gestionar la GO. HU: Publicar la GO. HU: Visualizar en la web en formato PDF la GO.	9 semanas
<b>Iteración # 2</b>	HU: Subir al sistema documento legislativo. HU: Listar Ediciones anteriores de la GO. HU: Listar documento legislativo por estados. HU: Cambiar el estado al documento legislativo. HU: Descarga en formato PDF de la GO. HU: Enviar notificación a los roles del cambio de estado. HU: Suscribirse al sistema. HU: Enviar por correo último número de la GO. HU: Eliminar la suscripción del sistema. HU: Gestionar Comentario. HU: Mostrar detalles de una gaceta oficial. HU: Imprimir la GO desde el sistema. HU: Verificar fortaleza de contraseña de usuario. HU: Cerrar sesión.	4 semanas

**Tabla 5: Plan de duración de las iteraciones.**

### 2.3.8 Plan de entrega

Después de confeccionar las HU, se realiza una estimación del esfuerzo necesario para marcar cuanto tiempo se demora la implementación de cada una de ellas, incluyendo las fechas de salida de las versiones funcionales del producto como se muestra en la tabla # 6.

Producto	Final de la 1era iteración Última semana de abril.	Final de la 2da iteración Primera semana de junio.
Sistema Web para la Edición y Publicación de la GO.	0.1	1.0

**Tabla 6: Plan de entrega.**

## 2.4. Arquitectura del sistema

El sistema está implementado sobre Drupal, por lo cual la arquitectura se hereda de este CMS. En lugar de considerar el sistema web como un conjunto de páginas interrelacionadas, Drupal estructura los contenidos en una serie de elementos básicos. Estos son los Datos o Nodos, Módulos, Bloques, Menú, Roles, Permisos, Plantillas o Temas. A continuación se muestra la descripción de cada una de las capas.

- Los **Datos** son los elementos básicos en que Drupal almacena la información, los contenidos. Así a medida que el sistema web crece, lo va haciendo el número de Nodos los cuales van formando un “depósito de Nodos” cada vez mayor.
- Los **Módulos** son los elementos que operan sobre los nodos y otorgan funcionalidad a Drupal permitiendo incrementar sus capacidades o adaptarlas a las necesidades de cada sitio web. Son complementos que se instalan en el sistema web proporcionándole nuevas funcionalidades.
- La siguiente capa de Drupal la constituyen los **Bloques y Menú**. Estos permiten estructurar y organizar los contenidos en la página web.
- La siguiente capa importante en Drupal es la de control de **Roles y Permisos**. Drupal dispone de un registro de usuarios y de roles que permiten especificar que tareas pueden realizar y a que contenidos puede acceder cada tipo de usuario.
- La última capa, es la capa de **Plantillas** y es la que establece la apariencia gráfica o estilo de la información que se le muestra al usuario. Esta separación entre información y aspecto gráfico permite cambiar el diseño u apariencia del sitio web sin necesidad de modificar los contenidos.

### Arquitectura Modular

El CMS Drupal está desarrollado en PHP siguiendo un modelo de programación estructurada y haciendo uso de un sistema de bases de datos relacional. Drupal es un sistema con una

Arquitectura Modular que permite ampliar sus funcionalidades a través de unos métodos uniformes de desarrollo e integración de nuevos módulos.

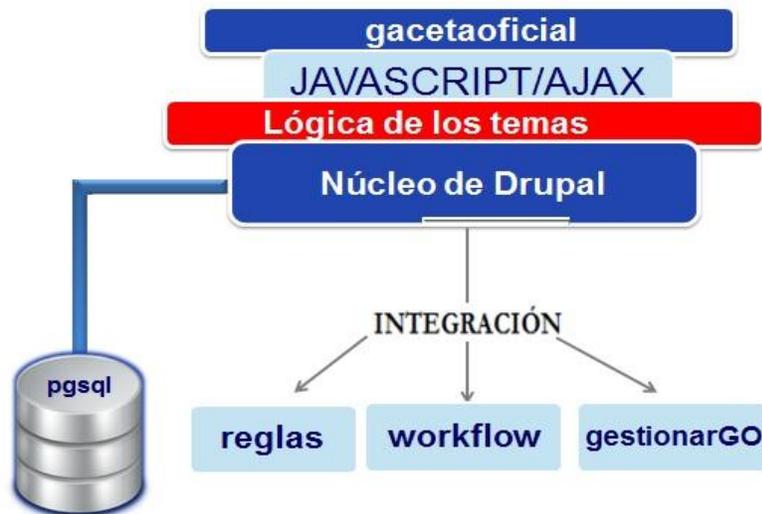


Figura 4: Arquitectura de Drupal (Rodríguez, 2012).

El núcleo de Drupal es el encargado de la integración de los módulos que sean instalados. Este le envía la información procesada de los módulos al componente de la lógica del tema que se encarga de la visualización de la información en las vistas del sistema. Finalmente los temas que se desarrollen utilizan los datos, permitiendo pre-procesar o alterar los mismos antes de ser mostrados. A su vez estos pueden utilizar las funciones del API de JavaScript y AJAX para brindar una interfaz más intuitiva a los usuarios (Rodríguez, 2012).

La propuesta de solución se basa en la utilización de módulos previamente incluidos por el núcleo de Drupal y otros que son contribuidos por la comunidad internacional de Drupal. Además, de desarrollar los propios módulos que son los que le darán soporte a la propuesta de solución.

### Descripción de los Módulos principales de la propuesta de solución:

#### Módulos Contribuidos de la Comunidad de Drupal utilizados:

**Workflow:** Permite definir el flujo o proceso por el cual transita un determinado nodo en un sistema en Drupal por ejemplo en la propuesta de solución es el documento legislativo mientras va cambiando por los diferentes estados.

**Views:** La acción de este módulo es definir páginas, temas dándole una vista detallada de cada elemento seleccionado.

**Rules:** Definir las reglas que se necesiten para un determinado problema, en este caso se define reglas para las notificaciones.

### Módulo Desarrollado:

**gestionarGO:** Complementar las funcionalidades del sitio utilizando los demás módulos contribuidos. Generar la plantilla que conformará la estructura del documento de la GO. Realizar la suscripción de los usuarios al sistema así como eliminarla.

## 2.5. Patrones de diseño

El CMS Drupal, en su versión 7.26, utiliza recursos de la Programación Orientada a Objetos como el manejo de objetos para datos, herencia, encapsulamiento, abstracción y polimorfismo. Las características estructurales que lo definen evidencian la aplicación de patrones de diseño (Drupal.org, 2012).

Drupal utiliza patrones de diseños propiamente orientados a objetos como los patrones Gang of Four (GoF), permitiendo la construcción de un diseño elegante y robusto. Los patrones GoF son clasificados según el propósito para el cual han sido definidos (Martin, 2009). Entre los patrones usados por Drupal se pueden citar los siguientes (Gamma, 2005):

### **Singleton (Instancia única)**

Este patrón se evidencia si se asocian los módulos y los temas como objetos, cada uno de ellos en si no almacena datos, sino que representan un conjunto de funciones que extienden el núcleo para añadirle funcionalidades al mismo, cada uno por separado. De esta forma se pueden considerar estos como una clase con una instancia singleton.

### **Decorator (Decorador)**

La evidencia de este patrón se encuentra en la definición de hooks por parte de los módulos del núcleo de Drupal y otros módulos contribuidos, que posibilitan que otros puedan extender el comportamiento de dichas funciones. De esta forma se brinda la flexibilidad de que nuevos módulos puedan modificar el comportamiento del núcleo en cuanto al tratamiento de los datos y en cada uno de los eventos del sistema. Este comportamiento podría ser imitado por el uso de decoradores, por ejemplo con el hook\_views\_exposed\_form\_alter definido por el módulo Views de Drupal, el módulo de la propuesta de solución implementa ese hook para alterar la presentación de los filtros de las vistas de búsqueda de gaceta.

### **Bridge (Puente)**

La capa de abstracción de datos de Drupal se encuentra implementada siguiendo el patrón Puente. Cada módulo debe ser programado de manera que sea independiente del motor de base de datos que utiliza el sistema. Esto se logra por la capa de abstracción de base de datos, sobre la que se pueden desarrollar nuevas siguiendo la API definida, para añadir soporte para nuevos sistemas gestores de base de datos, sin necesidad de modificar el código de los módulos.

### **Chain of responsibility (Cadena de responsabilidades)**

El sistema de menús de Drupal es la evidencia del patrón Chain of responsibility. En cada petición de una página, el sistema de menús de Drupal determina si hay algún módulo para responder la petición, si el usuario tiene acceso al recurso solicitado y que función se debe llamar para procesar la petición. En este proceso se transmite el mensaje de la petición por cada uno de los componentes que se encuentran inmersos. De esta forma se continúa la cadena hasta que un módulo atienda la petición, hasta que un módulo deniegue el acceso o hasta que la cadena se agote.

### **Command (Comando)**

Muchos de los hooks definidos por el núcleo de Drupal usan el patrón Command para reducir el número de funciones que son necesarias implementar, pasando la operación como parámetro conjuntamente con los argumentos. De esta forma se evidencia en el sistema de hooks, que cada módulo no necesariamente debe definir cada hook, sino solo los que tienen que ver con su propósito y funcionamiento.

## **2.6. Seguridad de la solución propuesta**

Con el propósito de garantizar la protección de los datos, Drupal implementa la seguridad a través de la autenticación de los usuarios utilizando sesiones y estableciendo un sistema de control de acceso basado en roles. La asignación de permisos se realiza a través de los roles del sistema, de esta forma el administrador no tiene que establecer los permisos para cada usuario, sino, se asignan los permisos a un determinado rol y se agrupan los usuarios por roles. Esta asignación por roles permite controlar a qué funcionalidades el usuario tiene acceso, de manera que la información accedida esté limitada y protegida en dependencia de los distintos niveles de usuarios, obteniendo como resultado un sistema más seguro. Otro mecanismo implementado por Drupal para brindar seguridad al sistema son las consultas parametrizadas que define la capa de abstracción de base de datos de Drupal, que evitan los ataques de

inyecciones SQL. Además Drupal cuenta con el módulo Database Logging el cual se encarga de guardar registros de la actividad de los usuarios en el sistema, lo que permite auditar los fallos que ocurran en el sistema. Otra forma de brindar seguridad es configurando los permisos de uso de los filtros tales como el “PHP Filter” y el “Full HTML”, de esta manera se logra evitar también las inyecciones SQL.

### 2.7. Diseño

La metodología de desarrollo de software XP sugiere que se realicen diseños simples y sencillos, para de esta forma lograr que sean de fácil entendimiento en la fase de implementación, lo que le costará menos tiempo al desarrollador llevar a cabo esta tarea.

#### 2.7.1 Tarjetas CRC (Cargos o Clases, Responsabilidad y Colaboración)

Las tarjetas CRC trabajan con la técnica de modelado basada en objetos, representando cada tarjeta CRC a un objeto, identificando las clases y sus responsabilidades. Las tarjetas están compuestas por el nombre de la clase colocado como título, en la parte izquierda se colocan las responsabilidades (funcionalidades) y en la parte derecha las clases que se implican en cada responsabilidad (Fuentes, 2008). La filosofía del CMS Drupal, no es basada en la Programación Orientada a Objetos (POO), adaptando la técnica de las tarjetas CRC a estas características, representando cada tarjeta CRC a un módulo, quedando las responsabilidades definidas como las funciones que realiza y los colaboradores serían módulos que se relacionan. A continuación se muestran la relación entre **Módulo-Responsabilidad y Colaboración** correspondiente al módulo que se implementó.

#### Módulo: gestionarGO

Responsabilidad	Módulos relacionados
Complementar las funcionalidades del sitio utilizando los demás módulos contribuidos.	<b>Workflow</b>
Generar la plantilla que conformará la GO.	<b>Rules</b>
Realizar la suscripción de los usuarios al sistema, así como el envío de correos con el último número de la GO.	<b>Views</b>

Tabla 7: Responsabilidad y colaboración del módulo gestionarGO.

### 2.7.2 Diagrama de paquetes

Los paquetes son elementos que permiten dividir un modelo en partes manejables mediante la agrupación de clases u otros paquetes, de ahí que se usen para reflejar la organización de paquetes y los elementos que los componen (Falgueras, 2008).

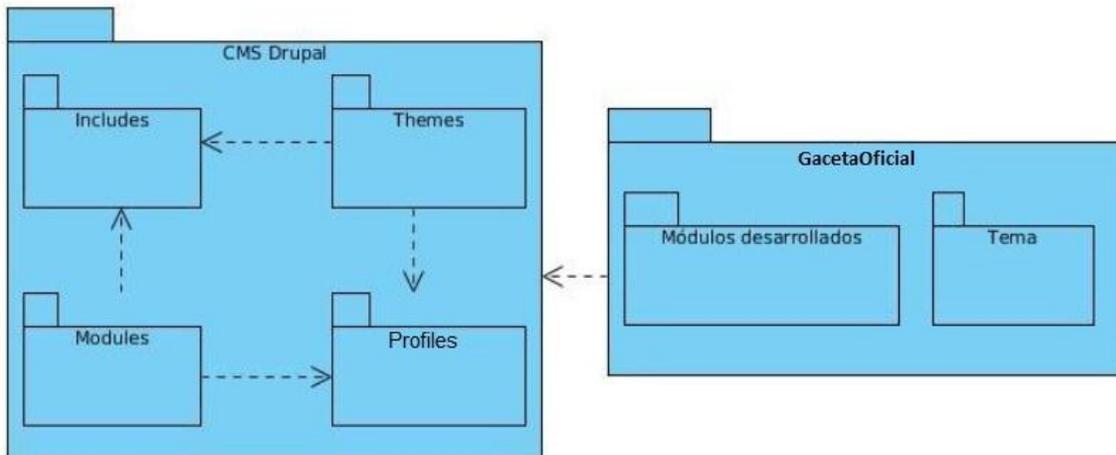


Figura 5: Diagrama de Paquetes del sistema.

En el diagrama se representa un paquete nombrado “CMS Drupal”, que incluye los paquetes contenedores de los componentes del CMS y otro paquete de los más generales es el denominado “GacetaOficial”, que incluye los elementos que conforman la aplicación. A continuación se describe el contenido de cada uno:

#### Paquete CMS Drupal:

**includes (incluidos):** Contiene un conjunto de ficheros indispensables para el funcionamiento de los módulos y el sistema Drupal, como las API de conexión a la base de datos.

**modules (módulos):** Incluye todos los módulos, que permiten las distintas funcionalidades del CMS, cuando desee incorporar un nuevo módulo sólo se tiene que copiar en dicha carpeta.

**themes (temas):** Incluye los temas que tiene de forma predefinida Drupal, permiten cambiar el estilo de la vista que se le muestra al usuario, cuando desee incorporar un nuevo tema sólo se tiene que copiar en dicha carpeta.

**profiles (perfiles):** Es un conjunto de instrucciones que permite definir qué módulos y opciones se quieren activar durante el proceso de instalación de Drupal.

### **Paquete GacetaOficial:**

**tema desarrollado:** Contiene el tema desarrollado para el sistema.

**módulos desarrollados:** Contiene todos los módulos que fue necesario implementar para el desarrollo de la aplicación.

Estos dos paquetes generales (CMS Drupal y GacetaOficial) lógicamente se encuentran relacionados, pues el paquete GacetaOficial se implementa haciendo uso de dicho CMS y por tanto utiliza las características y funcionalidades que este brinda.

### **2.8. Conclusiones parciales**

En el presente capítulo se realizó la caracterización del sistema a desarrollar que permitió especificar cómo se realizarán los mecanismos para eliminar los inconvenientes existentes en el sitio web actual, se realizó el modelo de dominio, el cual facilitó a los usuarios obtener los principales conceptos que se manejan en el desarrollo del sistema y cómo se realizan los procesos de edición y publicación electrónica. Se definieron las principales funcionalidades, recogidas en uno de los artefactos que propone XP, en este caso, las historias de usuarios; así como los requisitos no funcionales, los cuales brindan las cualidades que se deben tener en cuenta para desarrollar una solución adecuada. Se precisó el plan de iteraciones, el plan de entrega y la estimación de tiempo para lograr que las historias de usuarios se realicen en el tiempo considerado y la aplicación sea entregada en la fecha acordada. Además, se describe la arquitectura del sistema para entender su proceso de funcionamiento y los patrones de diseño con que cuenta Drupal posibilitaron obtener los conocimientos necesarios para construir un sistema con un diseño elegante y robusto.

## Capítulo 3: Implementación y Validación de la propuesta solución

En el siguiente capítulo se presentará el diagrama de componentes y la definición de los estándares de codificación que deben tener en cuenta los desarrolladores en la implementación del sistema web. Además se mostrará una descripción de las pruebas seleccionadas para validar la implementación realizada, y se dará a conocer los resultados obtenidos en la ejecución de cada una de ellas.

### 3.1. Diagrama de Componentes

Un componente es el empaquetamiento físico de los elementos de un modelo. El diagrama de componentes representa los tipos de elementos de software que forman parte de la confección de aplicaciones, dígame ficheros de datos, código fuente, librerías; y las dependencias entre ellos (Larman, 2003). Está conformado por componentes y paquetes, donde se muestra la estructura del sistema en términos de implementación a un alto nivel. La figura siguiente muestra el diagrama de componentes del sistema propuesto:

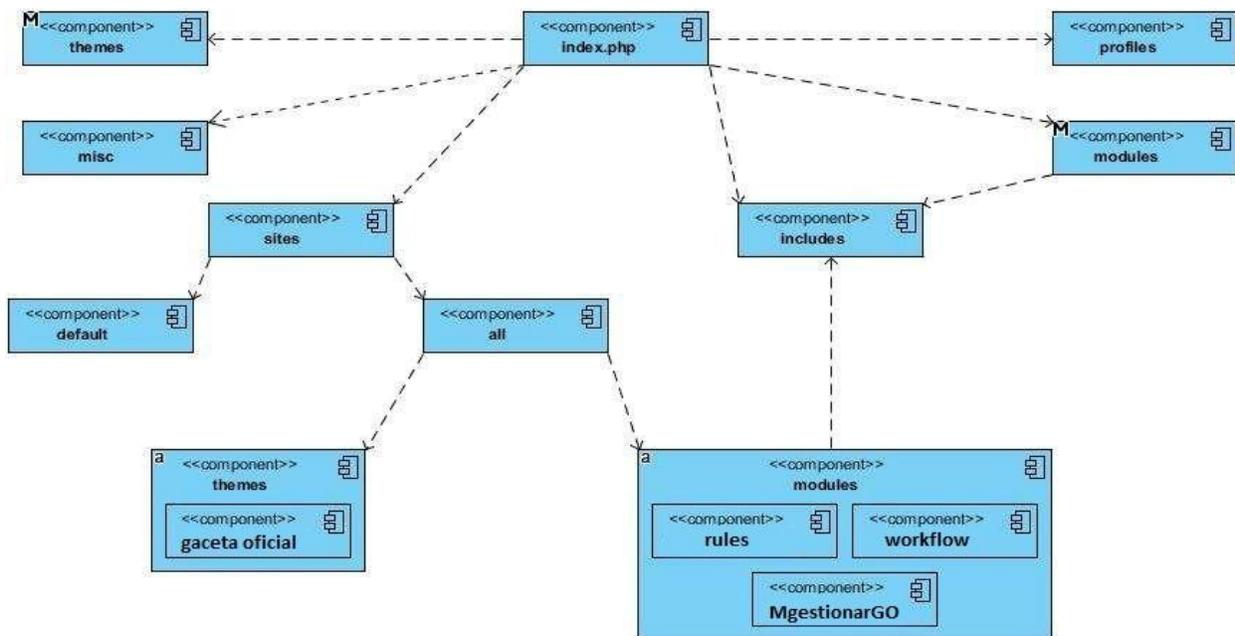


Figura 6: Diagrama de componentes general del sistema web para la Gaceta Oficial.

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

**index.php:** Este componente es el punto de inicio de la aplicación, a partir de esta entrada se solicitan los diferentes módulos del CMS Drupal.

**themes:** En este componente se incluyen los temas que vienen con la distribución de Drupal.

**misc:** Este componente incluye archivos javascript e imágenes requeridas por el sistema.

**includes:** Este componente tiene un conjunto de ficheros indispensables para el funcionamiento del CMS Drupal.

**modules:** En este componente se encuentran los módulos básicos del core de Drupal.

**profiles:** Este componente contiene los perfiles de instalación de Drupal.

**Sites:** La carpeta sites contiene los extras y modificaciones que se le añaden a la distribución original. En esta se encuentran los módulos adicionales creados, añadidos o descargados del repositorio de módulos de Drupal, colocados en **sites/all/modules**. Además los temas adicionales instalados o creados, en **sites/all/themes**.

### 3.2. Tareas de ingeniería

Las HU están divididas en distintas tareas, las cuales serán implementadas por el equipo de desarrollo, cada tarea es detallada especificando la HU relacionada, programador asignado, tiempo estimado, una breve descripción así como la fase en que se encuentra. Para una mejor organización a la hora de la implementación se dividieron en dos iteraciones según la complejidad de la HU.

#### Desarrollo de la iteración 1

En la primera iteración se tuvieron en cuenta aquellas HU de mayor importancia en cuanto a la funcionalidad que describen cada una, es decir, aquellas HU con prioridad alta.

#### Desarrollo de la iteración 2

En la segunda iteración se tuvieron en cuenta aquellas HU de menor importancia en cuanto a la funcionalidad que describen cada una, es decir, aquellas HU con prioridad media y baja.

Partiendo de la HU **Gestionar la GO** se establecen un grupo de tareas de ingeniería. Estas tareas se encuentran detalladas en el Anexo 1.

### 3.3. Estándares de código

Los estándares de codificación son un elemento fundamental en la implementación de un sistema, permitiendo que el código generado sea fácil de leer y modificar independientemente de quien ha sido su autor. Drupal le brinda a sus desarrolladores una serie de normas o pautas para fomentar el código de una forma común para todos. A continuación se describen los

## CAPÍTULO 3: IMPLEMENTACIÓN Y VALIDACIÓN DE LA SOLUCIÓN

---

estándares de codificación que establece Drupal y que fueron utilizados en la implementación de los módulos.

Para la declaración del código se tuvieron en cuenta los siguientes estándares (Gil, 2012):

- En Drupal la indentación debe ser de 2 espacios.
- Los operadores binarios, que se utilizan entre dos valores, deben separarse de estos valores, a ambos lados del operador, por un espacio. Por ejemplo, `$numero = 3`.
- Los operadores unarios como `++`, `--` no deben tener separación. Por ejemplo, `$numero++`.
- Cuando se escriba en PHP, siempre se deben utilizar las etiquetas `<?php` y `?>`, y en ningún caso la versión corta `<? y ?>`.
- En los ficheros `.module` y `.inc` se omite la etiqueta de cierre de PHP (`?>`).
- En los archivos de plantilla `.tpl.php`, cada fragmento de PHP debe llevar sus correspondientes etiquetas de apertura y cierre, para diferenciarlo del código HTML.
- Se pueden usar tanto las comillas simples ('cadena') como las comillas dobles ("cadena") para delimitar las cadenas de caracteres.
- La llave de apertura (`{`) se sitúa en la misma línea que la definición de la estructura, separada por un espacio.
- Se recomienda usar siempre las llaves `{}` aún en los casos en que no sea obligatorio su uso.
- Los nombres de las funciones deben estar escritos en minúsculas y las palabras separadas por guión bajo. Además, se debe incluir siempre como prefijo el nombre del módulo o tema, para evitar así duplicidad de funciones.
- El nombre de un módulo nunca debería incluir guiones bajos, aunque se componga de varias palabras.
- En su declaración, después del nombre de la función, el paréntesis de inicio de los argumentos debe ir sin espacio. Cada argumento debe ir separado por un espacio, después de la coma del argumento anterior. Ejemplo: `function forum_help($path, $arg) {`.
- Se pueden utilizar las etiquetas `/* */` para comentarios en varias líneas y `//` para comentarios de una única línea, suelen escribirse al principio de un archivo o de cada función. Se deben escribir frases completas, comenzándolas con mayúscula y terminándolas con un punto.

### 3.4. Diagrama de Despliegue

El diagrama de despliegue muestra la disposición física de los distintos nodos que componen el sistema. Un diagrama de despliegue modela la arquitectura en tiempo de ejecución de un sistema. Este muestra la configuración de los elementos de hardware (nodos) y muestra cómo los elementos y artefactos del software se instalan en esos nodos (Falgueras, 2008).

Se muestra a continuación el diagrama de despliegue correspondiente al sistema.

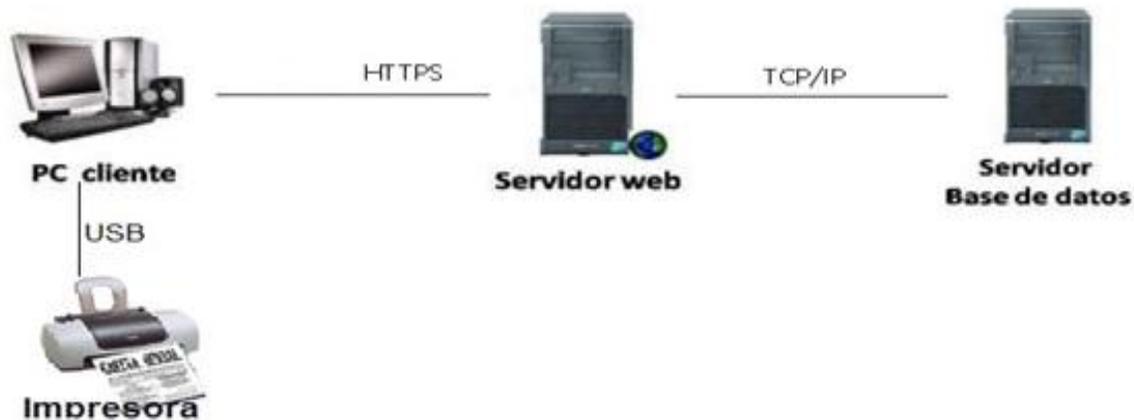


Figura 7: Modelo de despliegue.

#### Nodos

- ✓ **PC cliente:** Representa una computadora desde la cual el usuario accede a la aplicación.
- ✓ **Servidor web Apache:** Representa una estación donde se instala el servidor Apache sobre el cual se despliega la aplicación.
- ✓ **Servidor de Base datos PostgreSQL:** Representa el servidor donde se instala el SGBD (Sistema Gestor de Base Datos) PostgreSQL que responde a las peticiones hechas por la aplicación.
- ✓ **Impresora:** Representa una estación donde se puede imprimir la GO que el sistema genera para que sea distribuida.

#### Protocolos

- ✓ **TCP/IP** (Protocolo de comunicación y transmisión/Protocolo de Internet): Se utiliza en la comunicación entre el servidor y la base de datos para realizar operaciones sobre la información de las tablas.

✓ **HTTPS** (Protocolo seguro de transferencia de hipertexto): Es un protocolo de aplicación basado en HTTP, destinado a la transferencia segura de datos de hipertexto, para el proceso de autenticación por el motivo de que se envían los datos de los usuarios como el nombre y la contraseña y no se deben enviar sin encriptar por la red para evitar que se comprometa dicha información.

### 3.5. Validación

Seguidamente se describe el proceso de validación de los resultados obtenidos del proceso de implementación de la solución propuesta, con el objetivo de corroborar el correcto funcionamiento a partir de los requerimientos que se definen para el desarrollo de la solución.

Según **Pressman** las pruebas de software son un elemento crítico para la garantía de calidad del software y representa una revisión final de las especificaciones, del diseño y de la codificación (Pressman, 2002). Estas constituyen un conjunto de herramientas, técnicas y métodos que evalúan el desempeño de un programa. Además, involucran las operaciones del sistema, evaluando los resultados bajo condiciones controladas, lo que hace que la realización de pruebas al software sea un factor de vital importancia.

En el proceso de validación se realizaron las pruebas de aceptación o funcionales del sistema pues son la mejor forma de probar la aplicación de extremo a extremo: desde la petición realizada por un navegador hasta la respuesta enviada por el servidor. Las pruebas funcionales prueban todas las capas de la aplicación: el sistema de enrutamiento, el modelo, las acciones y las plantillas. En otras palabras, lo que se hace es probar un escenario correspondiente a la HU que se acaba de implementar en la aplicación.

### 3.6. Definición de la Estrategia de Pruebas

Para realizar las pruebas del software se aplican una estrategia moviéndose de adentro hacia afuera de la espiral como se muestra en la Figura 8. La **prueba de unidad** comienza en el vértice de la espiral y se centra en cada unidad del software, tal como está implementada en código fuente. La etapa avanza para llegar a la **prueba de integración**, donde el foco de atención es el diseño y construcción de la arquitectura del software. Otra vuelta hacia afuera se encuentra la **prueba de validación**, donde se validan los requisitos establecidos como parte del análisis de requisitos del software, comparándolos con el sistema que ha sido construido. Finalmente se llega a la **prueba del sistema** en la que se prueban como un todo el software y otros elementos del sistema.

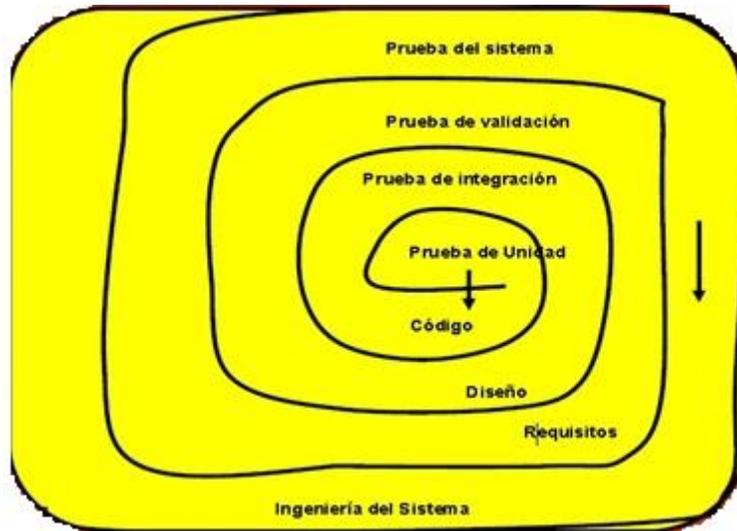


Figura 8: Estrategia de pruebas.

### Tipo de pruebas realizadas al sistema

Para la validación de los requisitos funcionales de la solución se define como tipo de pruebas a realizarse: las pruebas funcionales o de aceptación que estarán dentro de la parte de validación. Este tipo de pruebas funcionales están centradas en comprobar que las funcionalidades descritas en el documento de requisitos del sistema se cumplen con la implementación realizada (Oré, 2009). Por otra parte, para los requisitos no funcionales se establecen las pruebas de seguridad que tienen como objetivo hacer un análisis con el fin de encontrar fallos de seguridad tanto en el diseño como en la implementación de la aplicación y las de carga y estrés, las cuales permiten comprobar los límites operativos del sistema en condiciones de trabajo extremas, asegurando la calidad de respuesta del mismo ante este tipo de situaciones (Fraile, 2011). Se realizaron las pruebas de Usabilidad pues permiten realizar un análisis destinado a identificar y rectificar las deficiencias de usabilidad y errores existentes, que los usuarios pueden encontrar al usar una aplicación (Shawn, 2008).

### Método de prueba utilizado

Para la realización de las pruebas funcionales se tiene el método de prueba de caja negra. Las pruebas de caja negra son aquellas que se llevan a cabo sobre la interfaz del software. O sea, los casos de prueba pretenden demostrar que las funciones del software son operativas, que la entrada se acepta de forma adecuada y que se produce un resultado correcto, así como que la integridad de la información externa se mantiene, las cuales son realizadas desde el punto de

## CAPÍTULO 3: IMPLEMENTACIÓN Y VALIDACIÓN DE LA SOLUCIÓN

---

vista de interfaz de usuario, lo cual permite valorar el funcionamiento de la aplicación según la interacción del usuario con el sistema.

Las pruebas de caja negra intentan hallar errores tales como:

Errores de inicialización y de terminación.

Funciones incorrecta.

Errores de interfaz.

### Técnica de prueba utilizada: Casos de pruebas

El diseño de casos de pruebas para la partición equivalente se basa en una evaluación de las clases de equivalencia para una condición de entrada. Una clase de equivalencia representa un conjunto de estados válidos o inválidos para condiciones de entrada. Una condición de entrada es un valor numérico específico, un rango de valores, un conjunto de valores relacionados o una condición lógica.

Para garantizar la calidad del sistema se diseñaron pruebas funcionales que son aplicadas a través de pruebas de aceptación, en este tipo de pruebas se describen los escenarios para demostrar que una HU ha sido desarrollada correctamente. El principal objetivo de estas es asegurar que los requisitos han sido cumplidos satisfactoriamente y por lo tanto, el sistema es aceptable.

A continuación se describe el caso de prueba de aceptación que tiene relación con una de las HU descritas, realizado para verificar el correcto funcionamiento de la solución propuesta. Los demás casos de pruebas se encuentran en el Anexo 2.

<b>Caso de prueba de aceptación</b>	
<b>Código:</b> HU5_P1	<b>Historia de Usuario:</b> 5
<b>Nombre:</b> Realizar búsqueda avanzada de la GO.	
<b>Descripción:</b> Prueba para la funcionalidad.	
<b>Condiciones de Ejecución:</b> <ul style="list-style-type: none"><li>• Ninguno</li></ul>	

<b>Entrada/Pasos de ejecución:</b>		
Para ello se debe seguir los siguientes pasos:		
1. Entrar a la dirección donde se encuentra el Sistema de la GO.		
2. Seleccionar la opción “Búsqueda”.		
3. Seleccionar la opción “Búsqueda avanzada”.		
3. Insertar cualquiera de los siguientes criterios de búsqueda:		
<ul style="list-style-type: none"> <li>• Tipo de GO.</li> <li>• Organismo.</li> <li>• Año.</li> </ul>		
4. Dar clic en el botón Buscar.		
<b>Escenarios:</b>	<b>Resultados esperados</b>	<b>Evaluación de la prueba</b>
Insertar criterio de búsqueda correctamente.	Persona encontrada	Satisfactoria.
Campos vacíos.	Mensaje señalizando que debe insertar algún criterio de búsqueda.	Satisfactoria.
Criterio de búsqueda incorrectamente.	Mensaje señalizando que no se han encontrado resultados para el criterio de búsqueda insertado.	Satisfactoria.

**Tabla 8: Prueba de aceptación de la HU Realzar búsqueda avanzada de la GO.**

### **Resultados de las pruebas aceptación**

Se probaron 25 casos de prueba que responden a las funcionalidades del sistema web. En total fueron detectadas 15 no conformidades las cuales fueron resueltas, destacándose errores ortográficos y textos en idiomas diferentes del español.

Se realizaron tres iteraciones. En la primera iteración se detectaron 10 no conformidades, en la segunda iteración 5 no conformidades y en la tercera iteración no se detectaron no conformidades. Todas las no conformidades encontradas fueron resueltas por el equipo de desarrollo como se muestra en la figura 9.

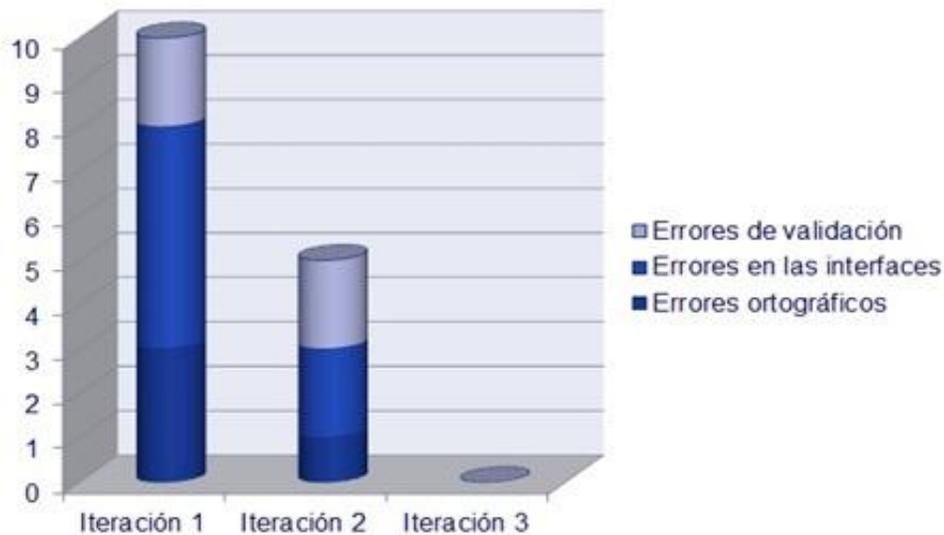


Figura 9: Resultados de la prueba de Aceptación.

### Seguridad

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

Las pruebas de seguridad fueron realizadas con la herramienta Websecurify, la cual es un entorno integrado para pruebas de seguridad web que proporciona la posibilidad de encontrar vulnerabilidades en los sitios web. También se realizaron estas pruebas mediante la lista de chequeo para prueba de seguridad (diseñadas por el Laboratorio Industrial de Pruebas de Software perteneciente a la empresa CALISOFT). Ver Anexo 3.

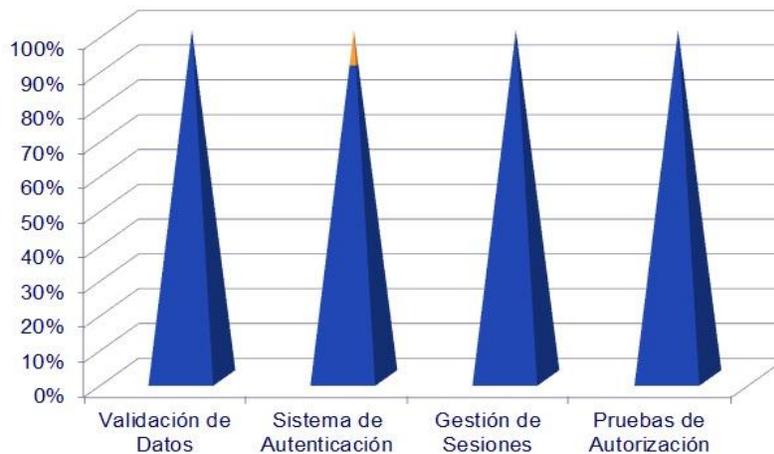
### Resultados de las pruebas de seguridad

Una vez, realizadas las pruebas con la herramienta WebSecurify al sistema web se obtienen el siguiente resultado.

Vulnerabilidad	Solución	Localización
Cross-site Request Forgery	Las URL y formularios que realizan operaciones importantes deben protegerse con testigos aleatorios (valores ocultos). Estos tokens se validan en el servidor antes de procesarla solicitud.	<a href="http://localhost/gaceta/busqueda-avanzada">URL:http://localhost/gaceta/busqueda-avanzada</a> . form:<form method="POST" action="http://localhost/gaceta/busqueda-avanzada?destination=busqueda-avanzada" en ctype="application/x-www-form-urlencoded" autocomplete="on"> ... </form>.

**Tabla 9: Resultados de la herramienta WebSecurityfy.**

El resultado después de realizar las pruebas mediante la lista de chequeo para prueba de seguridad indicó una buena seguridad del sistema como se muestra en la siguiente gráfica, 3 de 4 indicadores evaluados poseen un 100% de seguridad y 1 indicador posee un 92% de seguridad.



**Figura 10: Resultados de las prueba de Seguridad.**

### Carga y Estrés

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

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

### Resultados de las pruebas carga y estrés

Las pruebas de rendimiento fueron realizadas en la herramienta JMeter en su versión 2.9, diseñada para pruebas de carga de comportamientos funcionales y la medición del rendimiento. Prueba la resistencia y analiza el rendimiento en diferentes tipos de carga. El entorno en que fueron realizadas cumple con las siguientes características: 1 PC cliente, con un procesador Intel Core Dos Duo y 2GB de RAM.

A continuación se diseña el plan de pruebas de rendimiento para las diferentes acciones que los usuarios pueden realizar al conectarse al sistema web de la GO de la República de Cuba.

Para una muestra de 100 usuarios conectados concurrentemente, con un período de subida de un segundo (tiempo que espera cada usuario para realizar una petición) la aplicación generó los siguientes reportes:

Funcionalidad	URL	Máx.	Kb/s	Rendimiento	Avg. Bytes
Principal	localhost/gaceta/	54979	65.01	1.8/sec	37573.0
Ediciones Anteriores.	localhost/gaceta/gacetas-ediciones-antteriores	43013	64.80	2.3/sec	28992.0
Ediciones del Mes.	localhost/gaceta/gacetas-ediciones-del-mes	46739	65.00	2.1/sec	31757.0
Visualizar en línea la GO.	localhost/gaceta/ver-pdf/51	45300	87.63	2.2/sec	41548.0
Contáctanos.	localhost/gaceta/node/7	40572	66.61	2.4/sec	28310.0
Realizar búsqueda.	localhost/gaceta/buscar-gaceta?keys=	43675	62.34	2.0/sec	37740.0

Tabla 10: Resultados pruebas de rendimiento.

### Usabilidad

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

Las pruebas de usabilidad se realizaron mediante listas de chequeo (diseñadas por el Laboratorio Industrial de Pruebas de Software perteneciente a la empresa CALISOFT) relacionando el cumplimiento de aspectos incluidos en áreas o indicadores como se muestran en la figura 11. Para el cálculo del por ciento definido se responde una serie de preguntas en función de los aspectos analizados y se le otorga una evaluación, el mismo se evalúa de 1 en caso de mal y se le otorga el valor 0 en caso de que el elemento revisado no presente errores,

luego se calcula el porcentaje de preguntas evaluadas de bien en función del total y si el resultado es superior al 75 % se considera que la prueba arrojó resultados positivos.

El resultado de estas pruebas indicó una buena usabilidad del sistema, debido al cumplimiento mayor al 75% de los indicadores de cada área de usabilidad.

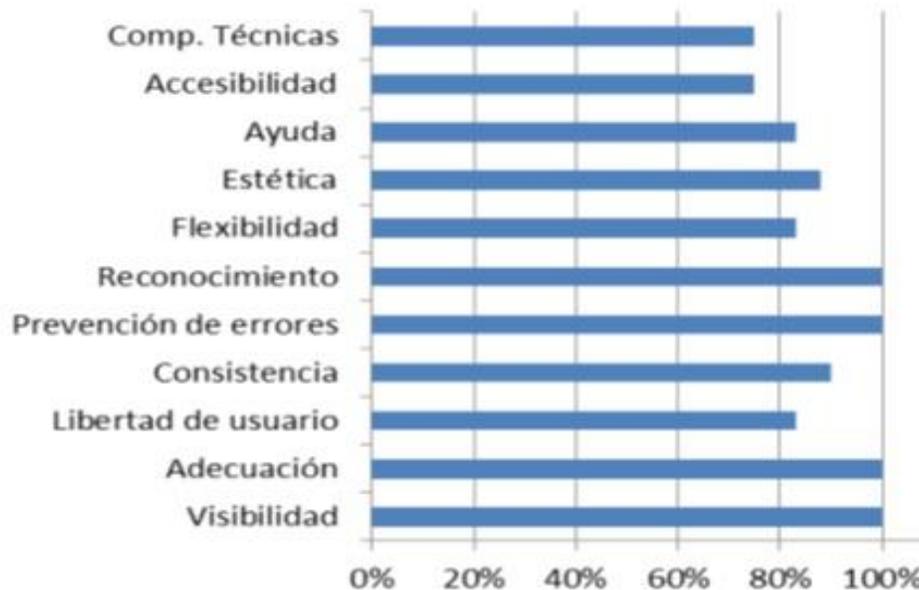


Figura 11: Resultados de la prueba de Usabilidad.

### 3.7. Validación de la Investigación

Para validar el objetivo general planteado es necesario medir el efecto que la variable independiente tiene sobre la variable dependiente. Esta medición es crucial para llegar a conclusiones importantes. Es por esto que los resultados de la medición deben ser suficientemente validados y confiables.

**Variable independiente:** desarrollo de un sistema web.

**Variable dependiente:** gestión de los procesos de edición y publicación electrónica de la GO.

Para hacer esta validación se realizó un caso de estudio aplicando el proceso de edición y publicación de los documentos legislativos de forma manual (Antes) y utilizando la herramienta informática para el mismo propósito (Después) y se compararon los resultados de cada uno de los indicadores en ambos casos. Los indicadores que se tomaron para evaluar la variable dependiente fueron: el tiempo, mecanismos de búsqueda, almacén de datos, administración de servicios, confidencialidad e integridad de la información y presentación del contenido. Los resultados se muestran en una tabla comparativa en el Anexo 4.

## CAPÍTULO 3: IMPLEMENTACIÓN Y VALIDACIÓN DE LA SOLUCIÓN

Luego de analizar y verificar que los resultados que se obtienen de la variable dependiente cumplen con los elementos planteados en el objetivo general y ofrece los mejores resultados para los indicadores analizados, se concluye que se encuentra correctamente validado. Por tanto en la presente investigación se cumplió con el objetivo general trazado inicialmente.

### 3.8. Pantallas de la aplicación

The screenshot displays the main page of the Gaceta Oficial de la República de Cuba website. At the top left is the logo for 'GACETA OFICIAL DE LA REPÚBLICA DE CUBA MINISTERIO DE JUSTICIA'. To its right, the ISSN number 'ISSN 1682-7511' and the text 'PUBLICACIÓN DIGITAL' are visible. A search bar with a 'Buscar' button is located in the top right corner. Below the header is a navigation menu with links for 'Inicio', '¿Qué es?', 'Historia', and 'Ayuda'. The main content area is divided into several sections: 'ENLACES EN PORTADA' (with links for 'Ediciones Anteriores', 'Ediciones del Mes', and 'Instrumentos jurídicos internacionales'), 'MUY CUBANO' (with 'Símbolos de la nación'), 'CONTACTOS' (with 'Información'), and 'QUIÉN ESTÁ CONECTADO' (showing 'Hay actualmente 1 usuario conectado.'). The central section features 'ÚLTIMA PUBLICACIÓN' (Gaceta No.: 30 Extraordinaria Especial, 2014/05/28, with a 'Descargar (68.54 KB)' link and 'Visualizar en línea' option) and 'INTERÉS JURISTA' (with a link to 'Decreto sobre normas para la publicación en la Gaceta Oficial'). Below this is 'PUBLICACIÓN EN PAPEL' (with 'De las suscripciones' and 'Tiendas' links). On the right side, there is a 'SUSCRIBIR A GACETA' section with an 'Enviar' button and radio buttons for 'Alta' and 'Baja', and a 'SITIOS AFINES' section listing 'Fiscalía General', 'Tribunal Supremo', 'Administración cubana', and 'Otros sitios de gacetas'. A footer contains the text 'Derechos Reservados - República de Cuba. Ministerio de Justicia - 2014' and an 'Encima' button.

Figura 12: Pantalla de la portada principal del sistema.

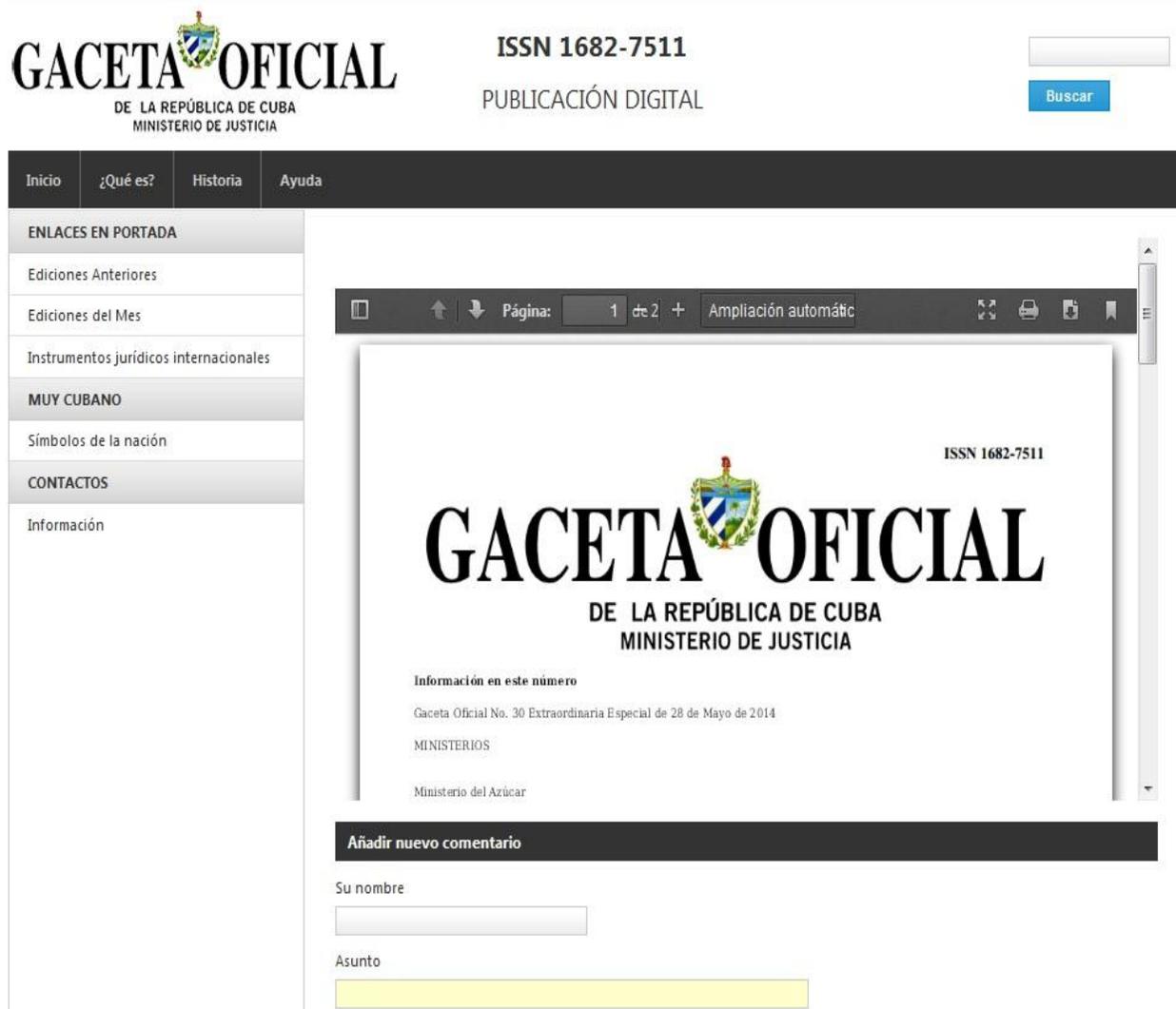


Figura 13: Pantalla de la funcionalidad Visualizar la GO.

### 3.9. Conclusiones parciales

En este capítulo la confección del diagrama de componentes ofreció una vista arquitectónica de alto nivel para ayudar al equipo de desarrollo en la implementación. Al mismo tiempo la definición de los estándares de codificación a tener en cuenta para la implementación del sistema web, permitió garantizar que el código posea alta calidad, menos errores y pueda ser mantenido fácilmente. Por otra parte la realización de las pruebas de software al sistema web facilitó encontrar los errores existentes para poder solucionarlos, posibilitando determinar y asegurar la calidad, robustez y fiabilidad del mismo. Con la culminación del capítulo se da por terminada la propuesta de solución relacionada al desarrollo de la aplicación para la edición y publicación electrónica de la GO de la República de Cuba.

## **Conclusiones generales**

Mediante la planificación de varias tareas al inicio de la investigación, se realizó un sistema web de forma exitosa y organizada, del cual se concluyen los siguientes aspectos:

El estudio de diversos sistemas en los diferentes ámbitos arrojó que no existe un sistema que realice los procesos de edición y publicación electrónica, por lo que a partir de ahí se defendió la idea propuesta y se desarrolló la presente investigación. La caracterización y comparación de varias herramientas y tecnologías ayudaron a seleccionar las más indicadas para la realización de la aplicación, lo que contribuyó a un correcto proceso de desarrollo. Con la identificación de los requisitos funcionales quedó explícito lo que el sistema debía hacer, los cuales llevados a historias de usuarios mostraron un mejor entendimiento acerca del funcionamiento del software, y con las condicionantes de los requisitos no funcionales se evidenció claramente las propiedades y cualidades del producto. A raíz de una arquitectura diseñada y otros patrones de diseño se logró la implementación de una aplicación escalable y a la medida, teniendo el usuario en todo momento el control sobre la aplicación. Para evaluar el comportamiento del sistema ante cada funcionalidad se realizaron pruebas de aceptación con el método de caja negra, donde los resultados fueron satisfactorios para un cien por ciento de todas las historias de usuarios planteadas.

Se logró el objetivo general de la investigación por lo que el sistema dio solución a las necesidades del cliente. La aplicación está lista para su utilización y con la capacidad de mejorar ampliamente en funcionalidades adaptándola a las condiciones del cliente que la solicite.

## Recomendaciones

Para futuras versiones del sistema web se recomienda:

- Brindar las posibilidades de navegar en el sistema web en distintos idiomas para lograr la internacionalización de las publicaciones de la GO.
- Profundizar en el estudio del CMS Drupal, pues cada día surgen nuevas actualizaciones de módulos y servicios que optimizarían el funcionamiento de la aplicación.
- Utilizar la solución propuesta en otros sistemas relacionados con la publicación electrónica de documentos.
- Aumentar las funcionalidades y servicios para lograr una mayor atención de los internautas, y su permanencia en el sistema web.

## Referencias Bibliográficas

**Aguiar, Mayelín Travieso. 2003.** Las publicaciones electrónicas: una revolución en el siglo XXI. [En línea] Diciembre de 2003. [Citado el: 07 de diciembre de 2013.] Disponible en: [http://www.bvs.sld.cu/revistas/aci/vol11\\_2\\_03/aci010203.htm](http://www.bvs.sld.cu/revistas/aci/vol11_2_03/aci010203.htm)

**Amoroso Fernández, Yarina.** Informática Jurídica: algunas consideraciones sobre la publicidad de las normas jurídicas. Memorias de la III Conferencia Internacional de Derecho e Informática de La Habana. Mayo 2000.

**AFFILIATES, O. C.** MySQL. Disponible en: <http://mysql.com>

**Brailstord, D.I. y R.I. Beach.** Electronic publishing. A journal and its production. Computer Journal 32(6): 482-493, Dec. 1989.

**bilib.es. 2012.** Estudio de los sistemas de gestión de contenidos web. [En línea] 13 de Junio de 2012. [Citado el: 29 de octubre de 2013.] Disponible en: [http://www.bilib.es/uploads/media/estudio\\_sistemas\\_gestion\\_contenidos\\_web\\_cms.pdf](http://www.bilib.es/uploads/media/estudio_sistemas_gestion_contenidos_web_cms.pdf)

**Camps Paré, Rafael. 2005.** Software Libre. Base de Datos. Introducción a las bases de datos. [En línea] Mayo de 2005. [Citado el: 29 de octubre de 2013.] Disponible en: [http://www.swcomputacion.f2s.com/Linux/007-Bases\\_de\\_datos.pdf](http://www.swcomputacion.f2s.com/Linux/007-Bases_de_datos.pdf)

**Carrillo Pérez, Isaías, Pérez González, Rodrigo and Rodríguez Martín, Aureliano David. 2008.** Metodologías de desarrollo de software. 2008.

**Centro de Apoyo Tecnológico a Emprendedores, Fundación Parque Científico y Tecnológico de Albacete.** Estudio de los sistemas de gestión de contenidos web. Análisis de las mejores soluciones de mercado (2012). 13 de Julio de 2012.

**Caballero Redondo, María José. 2012.** Análisis de aplicación: JMeter. [En línea] 17 de Marzo de 2012. [Citado el: 25 de Octubre de 2013.] Disponible en: [http://www.bilib.es/fileadmin/user\\_upload/analisis-bilibjvmeter-v0.2.pdf](http://www.bilib.es/fileadmin/user_upload/analisis-bilibjvmeter-v0.2.pdf)

**Drupal.org.** Drupal programming from an object-oriented perspective. [En línea] 29 de 06 de 2012. [Citado el: 12 de febrero de 2014.] <http://drupal.org/node/547518>

**developer.mozilla.org. 2011.** Mozilla Developer Network. ¿Qué es JavaScript? [En línea] 9 de Noviembre de 2011. [Citado el: 25 de octubre de 2013.] Disponible en: [https://developer.mozilla.org/es/docs/JavaScript/Acerca\\_de\\_JavaScript](https://developer.mozilla.org/es/docs/JavaScript/Acerca_de_JavaScript)

**drupal.org.es. 2012.** Comunidad de usuarios de Drupal. Comunidad de usuarios de Drupal. [En línea] 2012. [Citado el: 26 de Noviembre de 2013.] Disponible en: <http://drupal.org.es/caracteristicas>

**Espinosa, Julia y Dueñas, Lourdes.** Publicaciones electrónicas. Experiencias de diseño y desarrollo en el Centro de Información del CIDT. Congreso Internacional de Información 2002.

**Fuentes, Isidro.** Tarjetas CRC. [En línea] 28 de octubre de 2008. Disponible en: [http://informatica.escuelaedib.com/index.php?option=com\\_myblog&show="Tarjetas-CRC".html&Itemid=61](http://informatica.escuelaedib.com/index.php?option=com_myblog&show=)

**FALGUERAS, B. C.** Ingeniería del software 3ed. Barcelona: UOC, 2008. 41 p. ISBN 84-8318-997-6.

**Fraile, Luis. 2011.** Pruebas de rendimiento. [En línea] 2011. [Citado el: 8 de Mayo de 2014.] Disponible en: <http://www.globetesting.com/pruebas-de-rendimiento/>

**Flores Cueto, Juan José.** Método de las 6'D. UML - Pseudocódigo - Java. (Enfoque algorítmico). [En línea][Citado: 25 de octubre de 2013] Disponible en: <http://books.google.com/cu/books?id=IZkdLjZrcq4C&pg=PA39&dq=lenguaje+de+modelado+UML&hl=es&sa=X&ei=IUSIUZOBEsTc4AObvYHgCA&ved=0CGQQ6AEwCQ#v=onepage&q=lenguaje%20de%20modelado%20UML&f=false>

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

**Graf, Hagen, y otros. 2012.** Joomla! 2.5 Guía para principiantes. [En línea] 22 de Abril de 2012. [Citado el: 26 de octubre de 2013.] Disponible en: [http://comunidad.cav.uci.cu/index.php?option=com\\_phocadownload&view=category&id=27:manuales-y-tutoriales&Itemid=56](http://comunidad.cav.uci.cu/index.php?option=com_phocadownload&view=category&id=27:manuales-y-tutoriales&Itemid=56)

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

**Hechevarría Kindelan, Ángela.** Una propuesta de concepto y clasificación de publicaciones electrónicas y un análisis de su impacto en los profesionales de la información. Ciencias de la Información (Cuba): 28 (2), 97, junio, 1997.

**Hernández Violena. 2010.** Modelo de Negocio a través de Procesos (BPM).

**Highsmith, J.** Agile Software Development Ecosystems. (2002). Addison-Wesley Professional

**Hipp, D. Richard.** SQLite. [En línea] [Citado el: 29 de Octubre de 2013.] <http://www.sqlite.org>

**inei.gob.pe.** INEI. Instituto Nacional de Estadísticas Informáticas. Herramienta CASE. [En línea] [Citado el: 7 de octubre de 2013.] <http://www.inei.gob.pe/biblioineipub/bancopub/Inf/Lib5103/Libro.pdf>

**joomla.org. 2012.** What is Joomla? What is Joomla? [En línea] 28 de Septiembre de 2012. [Citado el: 29 de Octubre de 2013.] Disponible en: <http://www.joomla.org/about-joomla.html>

**Jacobson, Ivar, Booch, Grady y Rumbaugh, James. 2000.** El Proceso Unificado de Desarrollo de Software. [trad.] Salvador Canal y Carlos Sánchez. Madrid: Pearson Educación.S.A, 2000b. págs. 127-128. 84-7829-036-2.

**KABIR, M. J.** La biblia de Servidor Apache 2, 2003, Vol.: I

**Larman, Craig. 1999.** UML y Patrones. Introducción al análisis y diseño orientado a objetos. Luz María Hernández Rodríguez. México: PRENTICE HALL, 1999. pág. 536. 970-17-0261-1.

**Larman, Craig.** UML y Patrones. 2da Edición. s.l. : Prentice Hall, 2003.

**Luis Alarcón. LA.** [En línea] 10 de Abril de 2007. <http://luisalarcon.com/blog/?p=450>

**Marticorena, Juan R. 2012.** Drupal 7 para usuarios y clientes. [En línea] 2012. [Citado el: 29 de octubre de 2013.] Disponible en:  
[http://comunidad.cav.uci.cu/index.php?option=com\\_phocadownload&view=category&id=29:manuales-y-tutoriales&Itemid=56](http://comunidad.cav.uci.cu/index.php?option=com_phocadownload&view=category&id=29:manuales-y-tutoriales&Itemid=56)

**Martínez, Rafael.** 2010. PostgreSQL-es. [En línea] 2 de Octubre de 2010. [Citado el: 26 de Octubre de 2013.] Disponible en: <http://www.postgresql.org.es>

**mysql.com. 2012.** MySQL. [En línea] 29 de Septiembre de 2012. [Citado el: 26 de octubre de 2013.] Disponible en: <http://dev.mysql.com>

**Ministerio de Justicia de la República de Cuba. 1997** [En línea] 11 de Julio 1997 Gaceta Oficial de la República de Cuba [Citado el: 05 de diciembre de 2013.] Disponible en: [http://minjus.cu/html/legislacion\\_cubana.html.%20minjus.cu.%201682-7511](http://minjus.cu/html/legislacion_cubana.html.%20minjus.cu.%201682-7511)

**Martin, R. C. 2009.** “Principios y Patrones de Diseño”. [En línea] 2009. [Citado el: 15 de Febrero de 2014.] Disponible en:  
[http://www.dsi.fceia.unr.edu.ar/downloads/informatica/info\\_III/patrones.ppt](http://www.dsi.fceia.unr.edu.ar/downloads/informatica/info_III/patrones.ppt)

**Ministerio de Justicia de la República de Cuba. 1997.** Gaceta Oficial de la República de Cuba. [En línea] 11 de Julio de 1997. [Citado el: 26 de Abril de 2014.] Se encuentra disponible en [http://minjus.cu/html/legislacion\\_cubana.html](http://minjus.cu/html/legislacion_cubana.html) 1682-7511.

**Sánchez Fornaris Maite, Dayanis Alcantara Rabí. 2010.** Propuesta de una guía de métricas para evaluar el desarrollo de los Sistemas de Información Geográfica. [Online] Enero 4, 2010. [Citado el: 22 de Abril de 2014.] .

**Shawn Lawton, Henry. 2008.** Pruebas de usabilidad. Simplemente pregunta: Integración de la accesibilidad en el diseño. [En línea] 2008. [Citado el: 8 de Mayo de 2014.] Disponible en: <http://www.uiaccess.com/justask/es/ut.html>

**Martínez, Ander. 2009.** Apache JMeter. Manual de usuario v1.2. [En línea] 20 de Mayo de 2009. [Citado el: 8 de Abril de 2014.] Disponible en:  
<http://www.ejie.net/documentos/Herramientas/JMeter.%20Manual%20de%20usuario%20v1.2.pdf>

**netbeans.org. 2012.** NetBeans. [En línea] 2012. [Citado el: 27 de octubre de 2013.] Disponible en: [http://netbeans.org/index\\_es.html#](http://netbeans.org/index_es.html#)

**Olson, Philip. 2012.** Manual de PHP. [En línea] 2012. [Citado el: 25 de octubre de 2013.] Disponible en: <http://php.net/manual/es/index.php>

**Oliveros, Alejandro y del Valle Rojo, Silvana. 2012.** Requerimientos No funcionales para aplicaciones Web. [En línea] 2012. [Citado el: 1 de Marzo de 2014.] Disponible en: [http://www.41jaiio.org.ar/sites/default/files/432\\_ASSE\\_2012.pdf](http://www.41jaiio.org.ar/sites/default/files/432_ASSE_2012.pdf) ISSN: 1850-2792.

**Oré B., Ing. Alexander. 2009.** Pruebas funcionales. [En línea] 2009. [Citado el: 24 de Abril de 2014.] Disponible en: [http://www.calidadyssoftware.com/testing/pruebas\\_funcionales.php](http://www.calidadyssoftware.com/testing/pruebas_funcionales.php)

**Penadés, Carmen; Letelier, Patricio y Canós, Jase Hilario.** Metodologías Ágiles en el Desarrollo de Software. Metodologías Ágiles en el Desarrollo de Software. Alicante: s.n., 12 noviembre 2003.

**Pressman.** Ingeniería de Software, Un enfoque práctico. s.l. : McGraw-Hill Companies, 2002. 8448132149.

**Priolo, Ing. Sebastián.** Programación Extrema. [En línea] [Citado el: 15 de octubre del 2013.] [http://www.fcad.uner.edu.ar/jai/6JAI/XP\\_6JAI.pdf](http://www.fcad.uner.edu.ar/jai/6JAI/XP_6JAI.pdf)

**postgresql.org.** PostgreSQL 7.1 Tutorial Disponible en: <http://www.postgresql.org/files/documentation/pdf/7.3/tutorial-7.3.2-US.pdf>

**Ramakrishnan, Raghu y Gehrke, Johannes.** Database Management Systems. Berkeley: Osborne/McGraw-Hill, 2000. 0072440422.

**RODRÍGUEZ, F. G.** Experto en Drupal 7. Nivel avanzado. Forcontu S.L, 2012b. 43-44 p. ISBN 978-84-939410-5-5.

**Sarduy Domínguez, MsC. Yanetsys y Urra González, Lic. Pedro.** Sistemas de gestión de contenidos: En busca de una plataforma ideal. [En línea] 2006. [Citado el: 28 de octubre de 2013.] [http://bvs.sld.cu/revistas/aci/vol14\\_4\\_06/aci11406.htm](http://bvs.sld.cu/revistas/aci/vol14_4_06/aci11406.htm)

**Sabin-Wilson, Lisa.** WordPress for Dummies. Indianapolis, Indiana: Wiley Publishing, Inc., 2010. 978-0-470-59274-8.

**Serna M., Edgar. 2011.** Análisis y comparación de las propuestas recientes para diseñar casos de pruebas desde casos de usos orientados a verificar los aspectos del funcionamiento del software. [En línea] 2011. [Citado el: 26 de Abril de 2014.] Disponible en: [http://www.bdigital.unal.edu.co/6114/1/71617999.\\_2012.pdf](http://www.bdigital.unal.edu.co/6114/1/71617999._2012.pdf)

**The Electronic Encyclopedia.** 1988 Grolier Electronic Publishing, Inc. 1990, pp 27-32.

**w3.org. 2012.** W3C HTML. What is HTML? [En línea] 2012. [Citado el: 25 de octubre de 2013.] Disponible en: <http://www.w3.org/html>

**vyvquality.com. 2012.** Pruebas de seguridad. [En línea] 2012. [Citado el: 29 de Abril de 2014.] Disponible en: <http://www.vyvquality.com/w1/index.php/servicios/pruebas-de-seguridad.html> ISO 9001:200.

## Bibliografía consultada

**Amoroso Fernández, Yarina y MINJUS**, colectivo de Asesorías y Legislación del 1987-1995. resultados del proyecto MENJUR. 1987-2005.

**Amoroso, Yarina; Sass Manuela.** Textos constitucionales iberoamericanos. Sistema automatizado de recuperación de información sobre las Constituciones Iberoamérica, elaborado. Sociedad Cubana de Derecho e Informática. Pisa, Italia. 2000.

**Benavides, Ing. Adriel Alejandro Aliaga. 2011.** Complementos para el desarrollo de un portal web. [En línea] 22 de Junio de 2011. [Citado el: 26 de noviembre de 2013.] Disponible en: [http://www.informaticajuridica.com/trabajos/Complementos\\_desarrollo\\_portal\\_web.pdf](http://www.informaticajuridica.com/trabajos/Complementos_desarrollo_portal_web.pdf)

**Boris Beizer.** Software Testing Techniques. 2da. S.I.: Intl Thomson Computer, 1990. ISBN 978-1850328803.

**bilib.es. 2012.** Estudio de los sistemas de gestión de contenidos web. [En línea] 13 de Junio de 2012. [Citado el: 24 de octubre de 2013.] Disponible en: [http://www.bilib.es/uploads/media/estudio\\_sistemas\\_gestion\\_contenidos\\_web\\_cms.pdf](http://www.bilib.es/uploads/media/estudio_sistemas_gestion_contenidos_web_cms.pdf)

**Carrillo Pérez, Isaías, Pérez González, Rodrigo and Rodríguez Martín, Aureliano David. 2008.** Metodologías de desarrollo de software. 2008.

**Craig Larman, Prentice Hall.** El modelo de diseño. UML y Patrones 2ª Edición. 2003.

**Camarena Sagredo, Jesús Gamaliel, y otros. 2012.** Automatización de la codificación del patrón modelo vista controlador (MVC) en proyectos orientados a la Web. [En línea] 19 de Marzo de 2012. [Citado el: 25 de noviembre de 2013] Disponible en: <http://www.redalyc.org/articulo.oa?id=10423895005>

**Caldero Solís, Manuel.** Una explicación de la programación extrema (XP). Una explicación de la programación extrema (XP). [Online] 2003. <http://www.apolosoftware.com>

**Canós, José H., Letelier, Patricio y Penadés, María del Carmen. 2003.** Metodologías Ágiles en el Desarrollo de Software. [En línea] 2003. [Citado el: 25 de noviembre de 2013] Disponible en: [http://noqualityinside.com.ar/nqi/nqifiles/XP\\_Agil.pdf](http://noqualityinside.com.ar/nqi/nqifiles/XP_Agil.pdf)

**developer.mozilla.org. 2011.** Mozilla Developer Network. ¿Qué es JavaScript? [En línea] 9 de Noviembre de 2011. [Citado el: 26 de noviembre de 2013] Disponible en: [https://developer.mozilla.org/es/docs/JavaScript/Acerca\\_de\\_JavaScript](https://developer.mozilla.org/es/docs/JavaScript/Acerca_de_JavaScript)

**Date C. J: 2001.** Introducción a los Sistemas de Base de datos.2011. ISBN 0201-38590-2

**De Armas, Alejandro.2010.** Situación actual y perspectivas del desarrollo del Gobierno Electrónico en la Habana.

**Eridan Otto. 2009.** Proyecto Ley Chile: Acceso a las normas desde otros sistemas. Santiago de Chile: s.n., 2009.

**E-PING.2010.** E-PING ESTÁNDARES DE INTEROPERABILIDAD DE GOBIERNO ELECTRÓNICO. 2010.

**Ecured.** Ecured. Grupo de trabajo de Ecured. [En línea] [Citado el: 25 de abril de 2014.] [http://www.ecured.cu/index.php/Caso\\_de\\_Prueba](http://www.ecured.cu/index.php/Caso_de_Prueba)

**extremeprogramming.org. 2009.** Extreme Programming: A gentle introduction. [En línea] 2009. [Citado el: 3 de Diciembre de 2013.] Disponible en: <http://www.extremeprogramming.org>

**Fraile, Luis. 2011.** Pruebas de rendimiento. [En línea] 2011. [Citado el: 8 de Abril de 2014.] Disponible en: <http://www.globetesting.com/pruebas-de-rendimiento>

**Ferreras.** Aplicación de la usabilidad al proceso de desarrollo de páginas Web. Tesis de Máster. Madrid, España: Universidad Politécnica de Madrid, 2008.

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

**Gómez, Juan Carlos García. 2000.** Portal: definición, evolución y clasificación. [En línea] 2000. [Citado el: 26 de octubre de 2013.] Disponible en: <http://aprendeonline.udea.edu.co/lms/moodle/file.php/87/documentos/definicion-portales.pdf>

**Hispano,** Drupal. Características de Drupal. [En línea] 2005. [Citado el: 30 de Noviembre de 2013.] <http://drupal.org.es/caracteristicas>

**Lorenzo Pratts,** Documentos y Notas del curso impartido., Director del Departamento de Derecho Civil de la Universidad de Valencia. La Habana, 2000.

**Ministerio de Justicia de la República de Cuba. 1997.** Gaceta Oficial de la República de Cuba. [En línea] 11 de Julio de 1997. [Citado el: 26 de Abril de 2014.] Se encuentra disponible en [http://minjus.cu/html/legislacion\\_cubana.html](http://minjus.cu/html/legislacion_cubana.html).minjus.cu.1682-7511

**McKeever, S. 2003.** "Understanding Web content management systems: evolution, lifecycle and market". 2003.

**Márquez, José, Sampedro, Leonardo y Vargas, Félix, 2002,** Instalación y configuración de Apache, un servidor Web gratis. Red de Revista Científica de América Latina y el Caribe, España y Portugal [online]. Diciembre 2002. No. 012, p. 10–23. Disponible en: <http://redalyc.uaemex.mx/pdf/852/85201202.pdf>

**Nobrega, María De.** Herramientas CASE: Rational Rose. Curso Sistemas de Información II. [En Línea] [Citado el: 3 de noviembre de 2013.] [http://curso\\_sin2.blogia.com/2005/060401-herramientas-case-rational-rose.-por-maria-de-nobrega.php](http://curso_sin2.blogia.com/2005/060401-herramientas-case-rational-rose.-por-maria-de-nobrega.php)

**Nakano, R. 2001.** Web Content Management: A Collaborative Approach. Boston: Addison-Wesley, 2001.

**Noble, Ángela Martín y de Robert Biddle, James.** The XP Customer Role in Practice: Three Studies Agile. 2004.

**Pressman, Roger S. 2005.** Ingeniería de Software, un enfoque práctico. Quinta edición. 2005. pp. 281-303.

**P. Morville y L. Rosenfeld.** Information Architecture for the World Wide Web. S.I.: O'Reilly Media, 2006. ISBN 9781598299601.

**Rational Rose Data Modeler.** [En línea] [Citado el: 5 de noviembre de 2013.] <http://www.rational.com.ar/herramientas/rosedatamodeler.html>

**Rivera Alva, Eduardo. 2008.** Arquitectura de Software II. Diagramas de Componentes y Despliegue. [En línea] 11 de Noviembre de 2008. [Citado el: 5 de Abril de 2014.] Disponible en: <http://es.scribd.com/doc/7884665/Arquitectura-de-Software-II-Diagrama-de-Componentes-y-Despliegue>

**Sommerville, I. 2002.** Ingeniería de Software. [Online] Pearson Educación, 2002

**Timothy Berners-Lee:** World Wide Web Consortium [online]. 2009, Available from: <http://www.w3.org/DesignIssues/LinkedData.html>

**Visual Paradigm para UML.** [En línea] [Citado el: 24 de noviembre de 2013.] <http://www.software.com.ar/visual-paradigm-para-uml.html>

## Glosario de términos

- ✓ **API:** Acrónimo del inglés Application Programming Interface, traducido al español como Interfaz de programación de aplicaciones.
- ✓ **Aplicación:** En informática es un tipo de software diseñado para facilitar al usuario la realización de una determinada tarea o trabajo.
- ✓ **Aplicación web:** Se denomina aplicación web a aquellas aplicaciones que los usuarios utilizan accediendo a un servidor web a través de Internet o de una intranet mediante un navegador.
- ✓ **GNU/GPL:** Licencia orientada principalmente a proteger la libre distribución, modificación y uso de software. Garantiza a los usuarios finales (personas, organizaciones, compañías) la libertad de usar, estudiar, compartir (copiar) y modificar el software.
- ✓ **HTML:** HiperText Markup Language, por sus siglas en inglés, en español Lenguaje de Marcado de Hipertexto.
- ✓ **Ingeniería de software:** Disciplina o área de la informática que ofrece métodos y técnicas para desarrollar y mantener software de calidad.
- ✓ **JavaScript:** Lenguaje de script multiplataforma orientado a objetos, pequeño y ligero; no es útil como un lenguaje independiente, más bien está diseñado para una fácil incrustación en otros productos y aplicaciones, tales como los navegadores Web.
- ✓ **Librerías:** Conjunto de subprogramas utilizados para desarrollar software.
- ✓ **Módulo:** Es un software que agrupa un conjunto de subprogramas y estructuras de datos.
- ✓ **TCP/IP:** Se le denomina conjunto de protocolos TCP/IP, en referencia a los dos protocolos más importantes que la componen: Protocolo de Control de Transmisión (TCP) y Protocolo de Internet (IP), que fueron dos de los primeros en definirse.
- ✓ **Multiplataforma:** Término usado para referirse a los programas, sistemas operativos, lenguajes de programación, u otra clase de software, que puedan funcionar en diversas plataformas.
- ✓ **TICs:** Las Tecnologías de la Información y la Comunicación (TICs) son un conjunto de técnicas, desarrollos y dispositivos avanzados que integran funcionalidades de almacenamiento, procesamiento y transmisión de datos.