

# **Universidad de las Ciencias Informáticas**

**Título:** Sistema de Gestión Editorial para los medios de prensa cubanos v2.0.

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

**Autores:**

**Ively Viera Fajardo**

**Lázaro Raúl Falcón Rodríguez**

**Tutores:**

Ing. Yuneldis Reyes Velázquez

Ing. Gilberto Lissabet Hernández

MSc Dúnnia Castillo Galán

La Habana, 2015.

Año del 57 aniversario de la Revolución

## DECLARACIÓN DE AUTORÍA

Declaramos ser 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 \_\_\_\_\_.

Ively Viera Fajardo

\_\_\_\_\_

Autor

Lázaro Raúl Falcón Rodríguez

\_\_\_\_\_

Autor

Ing. Yuneldis Reyes Velázquez

\_\_\_\_\_

Tutor

MSc. Dúnnia Castillo Galán


\_\_\_\_\_

Tutor

Ing. Gilberto Lissabet Hernández

\_\_\_\_\_

Tutor



*Cualesquiera que hayan sido nuestros  
logros, alguien nos ayudó siempre a  
alcanzarlos.*

*Althea Gibson*

## DATOS DE CONTACTO

**Ing. Yuneldis Reyes Velázquez:** Graduado de Ingeniería en Ciencias Informáticas, Universidad de las Ciencias Informáticas, 2010. Especialista del Departamento de Servicios Informáticos para Internet (SENIT), Centro de Ideoinformática (CIDI), Facultad 1.

Correo electrónico: [yvelazquez@uci.cu](mailto:yvelazquez@uci.cu)

**Ing. Gilberto Lissabet Hernández:** Graduado de Ingeniería en Ciencias Informática, Universidad de las Ciencias Informáticas, 2009. Jefe del Departamento de Servicios Informáticos para Internet del Centro Ideoinformática de la Facultad 1.

Correo electrónico: [glissabet@uci.cu](mailto:glissabet@uci.cu)

**MSc. Dунnia Castillo Galán:** Licenciada en Periodismo, Universidad de Oriente, 2011. Jefa de la línea de Análisis de Información del Departamento de Operaciones Web y Análisis de Información (DOWAI), Centro de Ideoinformática (CIDI), Facultad 1. Máster en Estudios Interdisciplinarios de América Latina, el Caribe y Cuba (Universidad de La Habana, 2015).

Correo electrónico: [dunnia@uci.cu](mailto:dunnia@uci.cu)

## **AGRADECIMIENTOS**

*Siempre resultará difícil agradecer a todos aquellos que de una u otra manera me han acompañado en el desarrollo de este trabajo, porque nunca alcanza el tiempo, el papel o la memoria para mencionarlos, por tanto quiero agradecerles a todos ellos cuanto han hecho por mí.*

- ✓ *Primeramente le agradezco a mis padres por ser el pilar fundamental en todo lo que soy, por siempre creer y confiar en mí, por su apoyo incondicional en los momentos cuando dije no se puede, pero sobre todo por su amor incondicional:  
mamá, papá..... LOS QUIERO.*
- ✓ *A mis abuelo, tíos, tías y primos tan maravillosos que tengo, que nunca dejaron de confiar en mí, que han ayudado a lograr que mis ilusiones se conviertan en realidad.*
- ✓ *A Clau por siempre estar ahí en las buenas y en las malas aun cuando nos poníamos brava siempre supe que podría contar contigo para lo que fuera por eso hoy te agradezco mucho tu cariño de hermana porque en eso te convertiste desde el primer día, te quiero mi hermanita.*
- ✓ *A Lili por siempre estar ahí, por tu cariño y apoyo en las buenas y las malas y aunque me sacabas del paso algunas veces te convertiste en una gran amiga.*
- ✓ *A todos mis compañeros de aula, Suyín, Yanetsy, Orlando, Angel M, y claro al equipo 1 en todos los trabajos Rossangel, Osmel, Angel A. y Abel por pasar momentos inolvidables en estos 5 años*
- ✓ *A todos mis compañeros de aula, FEU y amistades que hice en estos 5 años.*

## *Sistema de Gestión Editorial para los medios de prensa cubanos v2.0*

- ✓ *A mis amigos de Cumbiamba, Espacio Abierto y claro de la Asociación Cultural de Amigos Peligrosos en especial a Kílmer, Dashiel, Lixán por ser mis confidentes en todos mis problemas y siempre estar ahí cuando los necesitaba, a Julio, Iván, Michel, Gretter, Yordanka, Exxon, Claudia, Michá, Ariagne, Jose, Yosleny, Jeiser, Angel Luis, Yení, Daniel, Asiel, Dayana, Carlos Jordan por convertirse en una familia para mí, por pasar momentos locos e inolvidables como el 17 de diciembre del 2014, ¿se acuerdan?, a todos gracias por su apoyo y amistad, nunca los olvidaré.*
  - ✓ *A Ama, Carlos y Dani, por todo su apoyo y cariño.*
  - ✓ *A mis tutores por su guía y apoyo en la realización de este trabajo.*
- ✓ *A mi compañero de tesis por su paciencia por darme la oportunidad de formar un gran equipo de trabajo donde reinó el apoyo y la consagración constante.*

*Ivelly Viera Fajardo*

- ✓ *Agradezco especialmente a mi madre por ser el motor impulsor de este gran sueño y por ser mi principal apoyo, sustento y fuerza para realizar todos mis sueños, a mi padre por proveerme un ejemplo a seguir y ser soporte vital de esta larga y bella tarea. A los dos por todos los sacrificios que hicieron para poder ver hoy cumplido este momento que también es de ellos.*
- ✓ *A mi abuela por darme tantos y buenos consejos y a mi tía por ofrecerme su amor y su apoyo incondicional. A mis primas, primos, tías, tíos y al resto de la familia por siempre creer en mí y porque siempre pude contar con su apoyo.*

## *Sistema de Gestión Editorial para los medios de prensa cubanos v2.0*

---

- ✓ *A los amigos y amigas que recolecté durante estos años y me permitieron pasar inolvidables momentos junto a ellos, a esos que verdaderamente se pueden llamar amigos y no necesitan ver su nombre en un papel para saber a quiénes me refiero.*
- ✓ *A mi compañera de tesis, por ser ejemplo de compañerismo y solidaridad y por su perseverancia y consistencia en la realización de la tarea más difícil de nuestra carrera.*
- ✓ *A nuestros tutores por guiarnos y apoyarnos durante el desarrollo de este trabajo.*

*A todos muchas gracias.*

*Lázaro Raúl Falcón Rodríguez*

## DEDICATORIA

*Se lo dedico a mi mamá por haberme apoyado en todo momento, por sus consejos, sus valores, por la motivación constante que me ha permitido ser una persona de bien, pero más que nada, por su amor. A mi papá por el valor mostrado para salir adelante, su apoyo incondicional y sobre todo por su amor. A mi familia y amigos que me dieron ánimos para seguir adelante aun cuando decía no se puede.*

*Ively Viera Fajardo*

*Se lo dedico a mi madre, a mi padre, a mi abuela y a mi tía, a ellos les debo ser quien soy y es por ellos que hoy estoy aquí. Los quiero mucho.*

*Lázaro Raúl Falcón Rodríguez*



## RESUMEN

Los Sistemas de Gestión Editoriales son una herramienta esencial para el eficiente flujo de trabajo de cualquier medio de prensa o publicación. Estos sistemas posibilitan la gestión de mayor cantidad de materiales periodísticos y la automatización de los principales procesos en la gestión editorial. El Sistema de Gestión Editorial para los medios de prensa cubanos en su versión 2.0, es una solución informática configurable y flexible que permite estandarizar el proceso editorial en un medio ya sea digital o impreso, describese este proceso como: creación, edición, aprobación y publicación de un material periodístico. Para su realización se emplearon herramientas como el Sistema Gestor de Contenidos Drupal 7.36 y el Sistema Gestor de Base de Datos PostgreSQL 9.3. Como lenguajes de programación se utilizaron PHP 5.3.8, HTML 5, CSS 3 y JavaScript 1.8, mientras que el servidor web empleado fue Apache 2.2.21.

**Palabras claves:** Sistemas de Gestión Editoriales, prensa, proceso editorial, Drupal.

## ÍNDICE

|   |           |
|---|-----------|
| <b>INTRODUCCIÓN .....</b>   | <b>1</b>  |
| <b>CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA DE LOS SISTEMAS DE GESTIÓN EDITORIALES .....</b>                                | <b>6</b>  |
| 1.1 SISTEMAS DE GESTIÓN EDITORIALES PARA LA PRENSA .....  | 6         |
| 1.1.1 Sistemas de Gestión Editorial en el mundo .....   | 7         |
| 1.1.2 Sistemas de Gestiones Editoriales en Cuba .....   | 9         |
| 1.2 ESTÁNDARES RELACIONADOS CON LA GESTIÓN DE MATERIALES PERIODÍSTICOS .....  | 10        |
| 1.3 SISTEMAS DE GESTIÓN DE CONTENIDOS PARA LA WEB .....   | 12        |
| 1.4 SISTEMAS DE GESTIÓN DE BASES DE DATOS .....   | 18        |
| 1.5 ANÁLISIS DEL SOPORTE TECNOLÓGICO PARA EL SISTEMA DE GESTIÓN EDITORIAL .....                                       | 20        |
| 1.6 HERRAMIENTAS PARA EL DESARROLLO DEL SOFTWARE .....  | 21        |
| 1.7 METODOLOGÍAS DE DESARROLLO .....  | 22        |
| 1.8 CONCLUSIONES PARCIALES.....   | 25        |
| <b>CAPÍTULO 2: ANÁLISIS Y DISEÑO DEL SISTEMA DE GESTIÓN EDITORIAL PARA LOS MEDIOS DE<br/>PRENSA CUBANOS V2.0.....</b> | <b>26</b> |
| 2.1 DESCRIPCIÓN DE LA PROPUESTA DE SOLUCIÓN .....   | 26        |
| 2.2 MODELO DEL DOMINIO.....   | 27        |
| 2.2.1 Descripción de Clases del Modelo del Dominio .....  | 27        |
| 2.2.2 Diagrama de Clases del Modelo de Dominio .....  | 29        |
| 2.3 CAPTURA DE REQUISITOS DEL SISTEMA DE GESTIÓN EDITORIAL PARA LOS MEDIOS DE PRENSA CUBANOS<br>V2.0.....             | 30        |
| 2.3.1 Lista de Reserva del producto (LRP) .....   | 30        |
| 2.3.2 Historias de usuarios y prototipos de interfaz .....  | 34        |
| 2.4 PATRÓN ARQUITECTÓNICO .....   | 40        |
| 2.4.1 Patrones de diseño.....   | 42        |
| 2.5 MODELO DE DISEÑO .....  | 44        |
| 2.5.1 Diagrama de Clases de Diseño (DCD) .....  | 44        |

|   |           |
|---|-----------|
| 2.6 CONCLUSIONES PARCIALES.....   | 46        |
| <b>CAPÍTULO 3: IMPLEMENTACIÓN Y PRUEBA DEL SISTEMA DE GESTIÓN EDITORIAL PARA LOS MEDIOS DE PRENSA CUBANOS V2.0.....</b> | <b>47</b> |
| 3.1 ESTÁNDAR DE CODIFICACIÓN .....  | 47        |
| 3.2 PLAN DE ENTREGA .....   | 48        |
| 3.3 MODELO DE BASE DE DATOS .....   | 49        |
| 3.3.1 Descripción del Modelo de Datos .....   | 50        |
| 3.4 DIAGRAMA DE COMPONENTES.....  | 52        |
| 3.5 DIAGRAMA DE DESPLIEGUE.....   | 53        |
| 3.6 PRUEBAS .....   | 54        |
| 3.6.1 Pruebas de Aceptación.....  | 54        |
| 3.6.2 Pruebas de Seguridad.....   | 58        |
| 3.6.3 Pruebas de Rendimiento .....  | 58        |
| 3.7 CONCLUSIONES PARCIALES .....  | 59        |
| <b>CONCLUSIONES .....</b>   | <b>60</b> |
| <b>RECOMENDACIONES.....</b>   | <b>61</b> |
| <b>REFERENCIAS BIBLIOGRÁFICAS.....</b>  | <b>62</b> |
| <b>GLOSARIO DE TÉRMINOS .....</b>   | <b>66</b> |
| <br><b>Índice de diagramas.</b>   |           |
| Diagrama 1. Diagrama de clases del Modelo de Dominio.....   | 30        |
| Diagrama 2. Diagrama de Clases de Diseño Escenario Crear Material Periodístico .....                                    | 44        |
| Diagrama 3. Diagrama de Clases de Diseño Escenario Editar Material Periodístico.....                                    | 45        |
| Diagrama 4. Diagrama de Clases de Diseño Escenario Eliminar Material Periodístico .....                                 | 45        |
| Diagrama 5. Diagrama de Clases de Diseño Escenario Listar Material Periodístico. ....                                   | 46        |

|   |    |
|---|----|
| Diagrama 6. Diagrama de base de datos ..... | 50 |
| Diagrama 7. Diagrama de Componentes .....   | 52 |
| Diagrama 8. Diagrama de Despliegue .....    | 54 |

## Índice de Imágenes.

|  |    |
|--|----|
| Imagen 1. Propuesta de interfaz del sistema. ....                    | 27 |
| Imagen 2. Arquitectura del CMS Drupal ( <i>VanDyk, 2011</i> ). ....  | 41 |
| Imagen 3. Comportamiento de las no conformidades por iteración. .... | 57 |

## Índice de Tablas.

|   |    |
|---|----|
| Tabla 1. Listado de Requisitos Funcionales. ....                        | 33 |
| Tabla 2. Lista Reserva de Productos (LRP). ....                         | 34 |
| Tabla 3. Descripción de la tabla <i>users</i> del Modelo de Datos. .... | 51 |
| Tabla 4. Descripción de la tabla <i>node</i> del Modelo de Datos. ....  | 51 |
| Tabla 5. Resultado obtenido para el escenario 1. ....                   | 59 |
| Tabla 6. Resultado obtenido para el escenario 2. ....                   | 59 |

## **Introducción**

El uso de las nuevas Tecnologías de la Información y las Comunicaciones (TIC) ha impulsado varios de los cambios que se han producido en la sociedad. No cabe duda que la irrupción de las TIC ha transformado la imagen del mundo, en especial, con la implantación de Internet, lo cual ha traído consigo el surgimiento de nuevas estrategias para el intercambio de información en formato digital. En este contexto, los Medios de Comunicación Masiva (MCM) posibilitan que grandes contenidos de información lleguen a diferentes lugares del planeta de forma inmediata.

Estos MCM cumplen entre otras funciones, la de informar, posibilitando que el acceso a la información resulte útil para aquellos que deseen estar al tanto del acontecer noticioso. “Estos medios de prensa, independientemente del tema de publicación o extensión, disponen de un flujo de trabajo que involucra a periodistas, editores, correctores, traductores y otro personal que varía en dependencia del medio” (*Serrano Paret, y otros, 2012, p 1*).

En el sector periodístico se conjugan dos factores de vital importancia: inmediatez y calidad en la publicación. En este sentido, los Sistemas de Gestión Editoriales y las plataformas de publicaciones web, juegan un papel protagónico al posibilitar la gestión de los materiales periodísticos tales como: noticias, crónicas, artículos, cartas, editoriales, entrevistas y reportajes. Además de la automatización de los principales procesos y flujos de trabajo: revisión, corrección, traducción, publicación, documentación, edición, conservación y aprobación.

Cuba, no queda exenta de esta dinámica y en los últimos 15 años se perciben cambios cualitativos y cuantitativos en los principales medios de prensa que forman parte de la industria mediática, en los que si bien de manera empírica se han asumido estos cambios, hoy pueden apreciarse deficiencias en cuanto a:

- La existencia de diferentes herramientas informáticas para un uso común.
- No se encuentran normalizados los principales procesos editoriales.
- No se utilizan estándares internacionales para la representación y transmisión de contenidos digitales.

- Existe débil integración entre sus componentes.
- No se socializan las aplicaciones.
- Los actuales Sistemas de Gestión Editoriales están pensados para la prensa escrita, no así para la digital.

La Universidad de las Ciencias Informáticas (UCI), como parte de la informatización de la sociedad cubana, ha investigado y desarrollado un Sistema de Gestión Editorial para los medios de prensa cubanos que, si bien ha servido para mejorar el flujo de trabajo de los materiales periodísticos, presenta deficiencias como: base tecnológica obsoleta, problemas de interfaz, no cumple con algunos de los requisitos definidos por el cliente y el flujo de trabajo se hace complejo al utilizarlo, por lo que dicha aplicación requiere hoy de una modernización, generalización en su base tecnológica, arquitectura y conceptualización, previendo una futura universalización del acceso a Internet y la necesidad de información y gestión eficiente del conocimiento.

A partir de lo anteriormente expuesto, se plantea el **problema de investigación**: ¿Cómo estandarizar el proceso editorial en los medios de prensa cubanos?

Se define como **objeto de estudio**: los Sistemas de Gestión Editoriales, y como **campo de acción**: los Sistemas de Gestión Editoriales para los medios de prensa cubanos.

La presente investigación tiene como **objetivo general**: Desarrollar un sistema informático que permita estandarizar el proceso editorial en los medios de prensa cubanos.

En virtud de dar solución a la propuesta, el objetivo general ha sido desglosado en los siguientes **objetivos específicos**:

- Elaborar una conceptualización teórica sobre los Sistemas de Gestión Editoriales.
- Realizar el levantamiento de los requisitos funcionales y no funcionales que deben estar contenidos en el Sistema de Gestión Editorial para los medios de prensa cubanos v2.0.

- Confeccionar el análisis y el diseño del Sistema de Gestión Editorial para los medios de prensa cubanos v2.0.
- Implementar el Sistema de Gestión Editorial para los medios de prensa cubanos v2.0.
- Validar el Sistema de Gestión Editorial para los medios de prensa cubanos V2.0 mediante pruebas de funcionalidad, rendimiento, seguridad y usabilidad.

Como **idea a defender** se plantea que: diseñando e implementando el Sistema de Gestión Editorial para los medios de prensa cubanos v2.0, se contribuirá a estandarizar el proceso editorial de los medios de prensa cubanos logrando normalizar el flujo de los materiales periodísticos generados en estos medios.

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

- Estudio y descripción de los Sistemas de Gestión Editoriales.
- Análisis de cómo funcionan los procesos de gestión de materiales periodísticos desarrollados en los medios de prensa cubanos.
- Determinación de los estándares relacionados con la gestión de materiales periodísticos.
- Elaboración de la propuesta de solución mediante los artefactos de la metodología de desarrollo seleccionada.
- Selección de las tecnologías y herramientas necesarias para el desarrollo del sistema.
- Definición de los elementos personalizables del sistema a desarrollar de acuerdo con las características de los medios de prensa cubanos.
- Desarrollo de las funcionalidades del Sistema de Gestión Editorial para los medios de prensa cubanos v2.0.
- Diseño y aplicación de pruebas de funcionalidad, rendimiento, seguridad y usabilidad al Sistema de Gestión Editorial para los medios de prensa cubanos v2.0.

## **Métodos de investigación utilizados:**

### **Métodos teóricos:**

**Analítico-Sintético:** Se utilizó en la recopilación y análisis de toda la información necesaria e importante sobre la gestión de materiales periodísticos y los sistemas de gestión editoriales, para de esta forma sintetizar la información que resultó realmente útil en el desarrollo de la investigación.

**Histórico-Lógico:** Se utilizó al inicio de la investigación para conocer los antecedentes e importancia de los Sistemas de Gestión Editoriales entendiendo su evolución y facilitando aprovechar puntos en común y conceptos teóricos relevancia y así poder conformar el marco teórico referencial.

**Modelación:** Se hace visible en el trabajo al generar los artefactos durante el proceso de ingeniería del software que ayudan a la comprensión y desarrollo del sistema.

### **Métodos empíricos:**

**Observación:** Permitió la recopilación atenta, racional, planificada y sistemática del fenómeno relacionado con el objeto de investigación; así como concretar las tecnologías y productos existentes para desarrollar el sistema deseado.

**Entrevista:** Proporcionó la obtención de la información necesaria para determinar las funcionalidades y características del sistema a desarrollar según las necesidades del cliente, permitiendo simplificar la obtención de los requisitos del sistema.

### **Justificación de la investigación.**

Los Sistemas de Gestión Editoriales se han convertido para cualquier medio de prensa en una herramienta necesaria para el desempeño eficiente de su flujo de trabajo. La relevancia de esta investigación radica entonces, en la importancia que se le atribuye al desarrollo de un Sistema de Gestión Editorial para los medios de prensa cubanos v2.0, que permitirá estandarizar el flujo de información de



estos, contribuyendo a brindar informes de producción, una capa de servicio a terceros sistemas que necesiten integrarse con él, así como perfeccionar el flujo de materiales periodísticos.

El documento está estructurado en los siguientes capítulos:

**Capítulo I:** *Fundamentación Teórica de los Sistemas de Gestión Editoriales.* Este capítulo tiene como objetivo realizar una investigación a la bibliografía sobre Sistemas de Gestión Editoriales, a través del estudio de diferentes definiciones y criterios de autores expertos en el tema, que permitieron desarrollar una base teórica sólida para el cumplimiento de esta investigación. Además, se hace referencia a las principales metodologías y herramientas utilizadas para el desarrollo web.

**Capítulo II:** *Análisis y Diseño del Sistema de Gestión Editorial para los medios de prensa cubanos v2.0.* Se determinan los requisitos funcionales y no funcionales. Además, se describe la arquitectura de software y los patrones arquitectónicos y de diseño que se utilizaron en el diseño e implementación de la solución propuesta. También se describen los prototipos de interfaz y otros artefactos que ayudaron al modelado del sistema a desarrollar.

**Capítulo III:** *Implementación y prueba del Sistema de Gestión Editorial para los medios de prensa cubanos v2.0.* Tiene como objetivo la implementación de las diferentes funcionalidades, elaboración de los diagramas de paquetes, componentes y despliegue. Además, se muestran los resultados de las pruebas realizadas al sistema durante todo el ciclo de desarrollo.

## **Capítulo 1: Fundamentación teórica de los Sistemas de Gestión Editoriales**

En este capítulo se desarrolla un estudio del estado actual y las tendencias de los Sistemas de Gestión Editoriales existentes en el mundo y en Cuba, así como el análisis de las herramientas, metodologías y estándares que serán necesarios para el desarrollo del sistema.

### **1.1 Sistemas de Gestión Editoriales para la prensa**

Actualmente existe poca bibliografía en cuanto a los Sistemas de Gestión Editoriales para la prensa debido a que son muy pocos utilizados por los mismos, es por eso que esta investigación estudia algunas de las particulares y consideraciones de distintos autores que se refieren a los Sistemas de Gestión Editoriales para publicaciones científicas en revistas, ya que sus características se asemejan a las de los medios de prensa.

“Un Sistema de Gestión Editorial (SGE) es un programa que ayuda a controlar, agilizar y hacer más eficiente el ciclo que tiene lugar desde que un periodista crea un material periodístico hasta que el mismo se publica o no, ya sea en un medio impreso o digital. Es por esto que su función está encaminada a la actividad más importante en un medio de prensa: la publicación de materiales periodísticos” (*Jiménez-Hidalgo, y otros, 2008, p 282*).

Los Sistemas de Gestión Editoriales de los medios de prensa son programas que permiten el envío en línea de los originales, la revisión de los mismos y el chequeo de las diferentes fases por las que transcurren las contribuciones. El desarrollo de estos programas informáticos favorece la revisión por pares utilizada en las publicaciones científicas. Estos sistemas, además de permitir la realización de diversas estadísticas; cuantificar la cantidad de materiales aceptados, rechazados, en proceso de revisión y listos para imprimir; también ayudan a detectar errores cometidos por los redactores, correctores y editores, y reducen de manera significativa el tiempo en el proceso de edición, además de facilitar la comunicación del autor con la editorial (*Las tecnologías de información en la actividad editorial: tendencias, contextos y perspectivas, 2009*)

Una vez analizado dicho criterio se asume, que un Sistema de Gestión Editorial es un programa que se encarga de optimizar el flujo de trabajo por el cual pasa un material periodístico, ya sean, periodistas,

editores, correctores, revisores, entre otros.

## **1.1.1 Sistemas de Gestión Editorial en el mundo**

En el mundo existe una amplia gama de software tanto de código abierto o de licencias libres, como de licencias comerciales o privadas, que permiten crear y gestionar el proceso editorial.

Entre las herramientas de código abierto o libre, encontramos:

**Digital Publishing System (DpubS):** “Es un software que facilita la organización, el suministro, la presentación y la publicación de las revistas científicas” (*Jeréz Abreu, y otros, 2010, p 26*).

Entre sus principales ventajas se encuentran:

- Diseñado para apoyar el cambio en el entorno de la comunidad académica.
- Está preparado para organizar congresos, jornadas o cursos y realiza la gestión de revistas.
- Apoya el creciente número y la variedad de publicaciones.

**Open Journal System (OJS):** “Sistema de gestión y publicación de revistas desarrollado por el *Public Knowledge Project* para mejorar el acceso a la investigación científica” (*Jeréz Abreu, y otros, 2010, p 27*).

Cuenta con diversas ventajas, las cuales se enumeran seguidamente:

- Herramienta totalmente web que puede ser accedida desde cualquier lugar mediante un navegador.
- Permite someter un artículo a revisión y soportar procesos de múltiples revisiones por parte de los árbitros.
- Apoya la labor de asignación de revisores/árbitro.
- Los autores pueden conocer el estado de los artículos enviados a revisión.

Entre las herramientas que contienen licencias privadas, podemos encontrar:

**KaliNews:** “Es una solución editorial de lenguaje de marcas extensible (XML por sus siglas en inglés) para

el manejo, transmisión y recepción de información editorial estructurada para diarios que imprimen ediciones regionales. Es una herramienta flexible que permite gestionar las informaciones asociadas al texto (metadatos) como: fecha, número de aparición, publicación, edición, formas autorizadas y nomenclaturas de palabras claves” (Experience,Soft, 2006). *KaliNews* presenta varias ventajas:

- Actualización automática de la configuración sobre terminales de trabajo de clientes remotos por comprobación del fichero de configuración XML.
- Cuatro vistas posibles para un artículo de prensa: informaciones y vista sintética, la historia integral, zoom sobre el texto y las fotos asociadas al texto.
- Control de elementos de texto y fotos asociadas con la historia y arreglo a las formas autorizadas.
- Contadores de signos por elemento de texto. Contadores de signos y de palabras seleccionadas.
- Calibración de texto y control de cada elemento de texto según forma seleccionada.
- Herramientas de búsquedas sobre el contenido de los artículos, sobre el conjunto de las rúbricas generales o locales, según el plan de clasificación.
- Redacción de texto en modo desconectado para los correspondientes remotos y periodistas independientes y transferencia directa al periódico:
  - Por cliente FTP integrado - Posibilidad de enviar a varios servidores FTP; transmisión de varios artículos y de las fotos asociadas en una sola operación con actualización de un registro de los envíos.
  - Por cliente SMTP integrado - Posibilidad de enviar un mensaje o un artículo a múltiples servidores SMTP (por correo electrónico - e-mails).

**Sistedit:** “Es un sistema editorial que usa las herramientas de *Adobe InDesign* y *Bridge* para la creación de productos editoriales. Automatiza tareas como el corte y tratamiento fotográfico, generación de PDF para impresión y XML para web para optimizar el proceso de producción y minimizar el tiempo del mismo” (Sur.ly, 2012).

## Ventajas:

- Multiedición.
- Corte y tratamiento de fotos automático.
- Integración con Adobe Bridge.
- Conducción durante el flujo de trabajo de un producto editorial.
- Control de usuarios.
- Control de producción.
- Generación de PDF para impresión.
- Generación de XML para Web.

Luego de haber investigado las características de varios de los Sistemas de Gestión Editoriales existentes, se concluye que aunque dichos sistemas no se ajustan a las características que se desean desarrollar debido a que estos son empleados en revistas electrónicas y diarios académicos, por lo que su flujo de trabajo es diferente al de los medios de prensa; aunque puede decirse que su estudio aportó ideas para desarrollar una solución, basado en algunas funcionalidades de los mismos.

### **1.1.2 Sistemas de Gestiones Editoriales en Cuba**

En la actualidad existen numerosas formas de realizar el proceso editorial en los medios de prensa del país, pues cada uno de ellos utiliza una manera particular para realizar este proceso. Influye en esto que muchos no cuentan con un sistema informatizado que les ayude en el manejo de toda la información que reciben y publican, haciéndoseles engorroso el trabajo; además de que en la mayoría de los casos de medios que poseen estos sistemas, fueron desarrollados por equipos de la propia institución adaptándolos a sus condiciones, por lo que de socializarlos no serían totalmente adaptables a las necesidades y características de los otros.

El Sistema de Gestión Editorial para el Periódico Granma (SGEPG) es uno de los mayores referentes de

este tipo, en uso en el país. Este es una aplicación informática sobre tecnología web que permite la correcta informatización del proceso de edición de materiales periodísticos en la sede de dicho periódico. Está basado en un Sistema de Gestión de Contenido (CMS por sus siglas en inglés) liberado bajo licencia libre y está desarrollado sobre el lenguaje PHP.

Por otra parte el grupo de desarrollo del periódico Juventud Rebelde trabaja su proceso de gestión editorial con una herramienta desarrollada en la misma entidad y que responde a las necesidades y características específicas de ese medio de prensa, pero de la cual no se pudo obtener la información tecnológica de su realización debido a que sus desarrolladores no lo permitieron.

Otro sistema utilizado en Cuba es el desarrollado por el Centro de Ideoinformática (CIDI) de la UCI, el cual permite ajustarse a la estructura del medio donde se instale y efectuar las operaciones que se realizan desde la creación de un material, hasta su publicación en la prensa plana o digital. Basado en el Sistema de Gestión de Contenidos Drupal y desarrollado sobre el lenguaje PHP, el mismo no presenta una interfaz intuitiva y muchos de sus requisitos presentan problemas en su implementación por lo que no responde a las necesidades del cliente, además fue desarrollado en el 2012 por lo que su tecnología se encuentra obsoleta.

Los sistemas mencionados anteriormente son productos exclusivos de cada medio por lo que no son extensibles a otros y presentan limitaciones para hacer frente a todo el flujo editorial de la prensa cubana la cual cada vez recibe y publica más información.

## **1.2 Estándares relacionados con la gestión de materiales periodísticos**

“Los medios de prensa diariamente producen un volumen considerable de información sobre disímiles acontecimientos. Esta información constituye soporte documental y materia prima para la construcción de los nuevos contenidos periodísticos, y su importancia radica en que son un constructo ideológico de la realidad en un momento dado y por tanto, el referente más consultado históricamente por disímiles usuarios” (*Debén González, Junio 2010*).

Gran parte de la información que transita por los sistemas de gestión editoriales para la prensa son trabajos periodísticos los cuales tienen la finalidad de comunicar e informar a los receptores sobre la

realidad nacional. Para un mejor estudio y trabajo, los mismos han sido segmentados en varios géneros periodísticos que responden a características específicas, aun cuando en la actualidad se está asistiendo a un proceso de unificación entre ellos que no permite en ocasiones segmentarlos con claridad.

Los géneros periodísticos, según Moreno Espinosa (2000, p 170), “reflejan el acontecer de un suceso y su interpretación, es decir, la información correspondiente y su comentario o valoración por parte del periodista”. Esta autora también clasifica los géneros periodísticos en dos tipos según su forma discursiva, los que dan a conocer hechos, que utilizan la forma expositiva, descriptiva y narrativa y los que dan a conocer ideas, que usan fundamentalmente la forma argumentativa.

Como resultado de la investigación realizada se seleccionaron los principales géneros periodísticos que se exponen a continuación.

## **Noticia**

“Es un texto que trata sobre asuntos de la actualidad informativa de interés general, con el fin de informar de forma objetiva, y cuyo texto se caracteriza por la claridad, sencillez y exactitud” (Yanes, 2002, p 251).

## **Entrevista**

Moreno Espinosa (2000, p 173) plantea que “la entrevista es el género mediante el cual un profesional de la información, el periodista, entra en contacto con un personaje público, el entrevistado, del que se presupone interés periodístico, bien por sus declaraciones, por su cargo o por su propia personalidad”.

## **Reportaje**

“El reportaje es un artículo en el que se plasma el resultado de las investigaciones hechas por el periodista sobre un tema de actualidad. El punto de partida de todo reportaje son los hechos que constituyen o constituyeron noticia y que mantienen vivo aún el interés.” (Pérez López, 2007).

## **Crónica**

“Es la información sobre unos hechos ocurridos durante un período de tiempo desde el lugar mismo o próximo a donde han ocurrido por un informador que los ha vivido como protagonista, testigo o investigador y que conoce las circunstancias que lo rodean” (*Moreno Espinosa, 2000, p 176*).

## El artículo

“Es el género periodístico que de manera personal interpreta, informa, analiza los acontecimientos y establece una tesis que luego constata. De los géneros de opinión, el artículo es quizá el menos confidencial de todos, del mismo modo que el menos actual. Se caracteriza visualmente por su extensión, que supera con creces a la de una” (*Moreno Espinosa, 2003*).

## El editorial

“Es el género que expresa el criterio del medio sobre los hechos más destacables. Ofrece el punto de vista institucional y, como consecuencia de ello, la redacción se ve afectada por un cierto protocolo, empleando un lenguaje menos personal. Suele tratar temas de eminente actualidad aunque no se limita a ellos. La finalidad de este género es la de intentar influir en la opinión pública” (*Moreno Espinosa, 2003*).

### 1.3 Sistemas de Gestión de Contenidos para la web

Un CMS permite crear, modificar y publicar contenidos web, especialmente portales web. Según Robertson (2003) la funcionalidad de un sistema de gestión de contenido se pueden dividir en varias categorías: creación, gestión, publicación y presentación de contenido, las cuales se explican seguidamente.

#### Creación:

Casi todos los sistemas de gestión de contenidos proporcionan un entorno de creación basado en la web, lo que simplifica aún más la aplicación, y permite la actualización de contenido para acceder de forma remota. Además, le ayuda a los creadores a concentrarse en el contenido para definir el formato de las páginas, la estructura, el aspecto visual, entre otros. Proporciona un editor WYSIWYG (*What You See Is What You Get*), o sea, lo que ves, es lo que se obtienes.

#### Gestión:



Una vez que una página ha sido creada, se guarda en un repositorio central, el cual almacena todo el contenido de la página, junto con los otros detalles de apoyo. Este repositorio central permite una amplia gama de funciones útiles que deberá facilitar el CMS:

- Hacer un seguimiento de todas las versiones de una página, qué cambió y cuándo.
- Asegurar que cada usuario sólo puede cambiar la sección del sitio que son responsables.
- La integración con las fuentes de información existentes.

### **Publicación:**

Una vez que el contenido final se encuentra en el repositorio, entonces se puede publicar, ya sea en la página web o Intranet. Los sistemas de gestión de contenidos cuentan con motores editoriales potentes que permiten el diseño y la apariencia de la página del sitio que se aplicará de forma automática durante la publicación. También puede permitir que el mismo contenido se publique en varios sitios. Por supuesto, cada sitio tiene un aspecto diferente, por lo que el CMS permite a los diseñadores gráficos y desarrolladores web especificar la apariencia que se aplica por el sistema. Esto permite a los autores concentrarse en escribir el contenido, dejando el aspecto del sitio en su totalidad al CMS.

### **Presentación:**

El CMS permite gestionar automáticamente la accesibilidad de la aplicación, con soporte de normas internacionales de accesibilidad como WAI (*Web Accessibility Initiative*) y adaptarse a las preferencias y necesidades de cada usuario. También puede proporcionar compatibilidad con los diferentes navegadores disponibles en todas las plataformas (Windows, GNU/Linux, Mac, Palm) y su capacidad de internacionalización le permite adaptarse al idioma, sistema de medidas y cultura del visitante.

Por todo lo expuesto anteriormente se pretende usar un CMS en el desarrollo del Sistema de Gestión Editorial para los medios de prensa cubanos v2.0. Pero antes de decidir cuál CMS utilizar se tuvo en cuenta los principales objetivos del sistema a desarrollar, según el público al que estará dirigido. Robertson (2002) se refiere a varias de las características que poseen los CMS que se consideran importantes en la implementación del sistema, las mismas son:

**Código abierto:** Tiene que ser de código fuente libre.

**Arquitectura técnica:** Tiene que ser fiable y permitir la escalabilidad del sistema para adaptarse a futuras necesidades con módulos. Es recomendable que se utilicen hojas de estilo (CSS por sus siglas en inglés) y patrones de páginas.

**Grado de desarrollo:** Disponibilidad de módulos que añadan funcionalidades.

**Soporte:** Debe tener soporte tanto por parte de los creadores como por otros desarrolladores, garantizando que en el futuro habrá mejoras de la herramienta y se podrán encontrar respuestas a los posibles problemas.

**Posición en el mercado y opiniones:** Asegurar que la herramienta a desarrollar pueda reutilizarse en el futuro, además de que las opiniones de los usuarios y los expertos sobre estas sean favorables.

**Usabilidad:** Fácil de usar y aprender, ya que los usuarios no siempre serán técnicos por lo que hay que asegurar que la herramienta sea utilizada sin muchos esfuerzos con el máximo rendimiento.

**Accesibilidad:** Tendría que cumplir un estándar de accesibilidad, actualmente el más extendido es WAI.

**Velocidad de descarga:** Teniendo en cuenta que no todos los usuarios disponen de líneas de alta velocidad, las páginas se tendrían que cargar rápidamente.

**Funcionalidades:** No se espera que todas las herramientas ofrezcan todas las funcionalidades, ni que éstas sean las únicas que tendrá finalmente la *web*.

De acuerdo a estos criterios se decide hacer una comparación entre los siguientes CMS: Wordpress, Drupal y Joomla.

## **Wordpress:**

Es uno de los CMS “enfocado en la creación de *weblogs* que está bajo la licencia GPL (*General Public License*). Las causas de su enorme crecimiento son, entre otras, su licencia, su facilidad de uso y sus

características como gestor de contenidos. Está basado en estándares y posee todos los servicios para crear una verdadera comunidad de *weblogs*” (Macias Rodríguez, y otros, junio del 2007, p 42).

Características de WordPress:

- Gestor de contenidos desarrollado en PHP y MySQL.
- Fácil instalación, actualización y personalización.
- Separa el contenido y el diseño en XHTML y CSS.
- Usa plantillas para generar las páginas dinámicamente.
- Utiliza temas disponibles y puede crear y compartir sus propios temas.
- Es un sistema extensible y configurable.
- Protege los artículos mediante contraseñas.
- Soporta de manera total las especificaciones RSS 1.0, RSS 2.0 y ATOM. El formato RSS 2.0 es soportado con marcas para archivo, de manera que agregar archivos MP3 es muy fácil.

**Joomla:**

“Potente gestor de contenidos web que permite crear sitios web elegantes, dinámicos e interactivos. Por su diseño, potencia, flexibilidad y por sus enormes posibilidades de ampliación se está convirtiendo en el sistema de publicación preferido por muchos centros educativos y por millones de *webmasters* en todo el mundo para desarrollar su portal web” (Barriocanal, 2009).

Este CMS ofrece una gran variedad de características y según Barriocanal (2009) estas son las más importantes:

- Insertar y administrar la publicación de anuncios gráficos o banners.
- Realizar encuestas en línea a los visitantes o usuarios de la web.

- Distribuir noticias en formato RSS.
- Sindicar noticias externas procedentes de otros sitios que dispongan de un canal RSS.
- Habilitar formularios de contacto para que visitantes y usuarios puedan enviar mensajes a los administradores y otros usuarios de la web.
- Enviar con un solo clic mensajes masivos a todas las personas registradas en el sistema.

## Drupal:

Según lo expresado en el *Estudio de los Sistemas de Gestión de Contenidos Web. Análisis de las mejores soluciones del mercado (2012, p 20)* un sistema de administración de contenidos *web* es especialmente versátil. No está dirigido a un tipo de escenario específico, ya que el límite de este CMS lo impone el desarrollador; al igual que ocurre con muchos otros CMS. Son muchas las características que sitúan a Drupal entre los CMS más destacados del mercado, estas son:

- Dispone de un entorno de personalización robusto. Tanto el contenido como la presentación pueden ser tratados de forma individual de acuerdo a unas preferencias definidas por el usuario. La gestión de contenido se realiza como objetos independientes, de forma que puede realizarse un tratamiento individualizado de la información, facilitando su inclusión en cualquier página o permitiendo comentarios específicos sobre cada uno de ellos.
- Los mecanismos de actualización de contenidos son realmente sencillos, permite editar la mayor parte de los contenidos tanto desde el *frontend* como desde el *backend*
- Ofrece la posibilidad de gestionar las taxonomías y la estructuración de contenidos de forma personalizable, algo indispensable para sitios de complejidad media y alta.
- Desde el punto de vista de la seguridad, la gestión de permisos se destaca por encima de cualquier otra característica; ofrece un sistema muy avanzado y completamente personalizable a nivel de rol y páginas.

- El rendimiento y la escalabilidad son otras de sus señas de identidad: sistema de caché avanzado, replicación de base de datos, balanceo de carga, mecanismos de control de congestión configurable para habilitar o deshabilitar módulos, etc.
- La comunidad de desarrolladores es otro de los puntos fuertes de Drupal, ofreciendo un desarrollo dinámico y un soporte amplio basado en foros *Web*.
- Dispone de cientos de extensiones, las cuales se encuentran agrupadas según funcionalidad en distintas categorías: administración, control de acceso, eventos, comercio, comunidad, contenidos, gestión de usuarios, búsquedas, etc.

Con respecto a las características técnicas, debe mencionarse que Drupal se encuentra liberado bajo licencia GPL y utiliza PHP como lenguaje de programación, MySQL como motor de base de datos, aunque también puede funcionar con PostgreSQL o SQLite y Apache o Microsoft IIS como Servidor *Web*.

De manera general los tres CMS son intuitivos y sencillos de gestionar. Para obtener una web simple todas son buenas herramientas pero cuando se trata de manejar componentes o utilidades más específicas como puede ser *Jdownloads*, en *Joomla* no hay color, porque este ofrece más funcionalidades y recursos para encontrar y adaptar componentes o módulos más específicos.

En Drupal está muy arraigado el concepto de seguridad desde el mismo núcleo. De hecho módulos del propio Drupal, vienen limitados en cuanto a visibilidad para usuarios anónimos por defecto. Pero realmente los tres CMS permiten dotar de roles y seguridad con la misma sencillez y robustez a cualquier parte de nuestro sitio.

Los tres permiten trabajar desde el *front-end*, pero el que más fácil y rápido movimiento permite para las modificaciones es Drupal.

Según los CMS estudiados se puede concluir que Drupal es el adecuado para desarrollar el Sistema de Gestión Editorial, utilizado en grandes y medianos proyectos, donde la estabilidad, escalabilidad y la potencia sean más importantes que la facilidad de uso y la estética, características estas, indispensables

para el tipo de sistema que se quiere desarrollar. Además es un CMS de nivel empresarial altamente recomendado para su utilización en instituciones periodísticas.

También se tuvo en cuenta que los sistemas creados anteriormente por la universidad fueron implementados en Drupal y que el equipo de desarrollo poseía experiencia en la utilización de este CMS, el cual es flexible, potente y una buena opción para realizar sitios más complejos, aunque requiere un mayor trabajo y contar con amplios conocimientos del mismo. Permite definir reglas muy detalladas sobre dónde debe mostrarse el contenido y construye sus propios tipos de contenido personalizado. Cuenta con una amplia variedad de funcionalidades para la comunidad, contenido para usuarios presentados, así como una gran diversidad de estructuras. Cabe destacar que en cuanto al flujo de trabajo, Drupal tiene mejor aceptación que Joomla y WordPress. Por tanto la investigación centralizará su estudio en el CMS Drupal para extender sus funcionalidades hacia el desarrollo del Sistema de Gestión Editorial para los medios de prensa cubanos v2.0.

## 1.4 Sistemas de gestión de bases de datos

“Es el software que permite la utilización y/o la actualización de los datos almacenados en una o varias bases de datos por uno o varios usuarios desde diferentes puntos de vista. Su objetivo fundamental consiste en suministrar al usuario las herramientas que le permitan manipular, en términos abstractos, los datos, o sea, de forma que no le sea necesario conocer el modo de almacenamiento de los datos en la computadora, ni el método de acceso empleado” (Martínez Bravo, 2007). Dentro de los gestores más conocidos y utilizados están MySQL y PostgreSQL, este último de código abierto.

### MySQL

“Es, sin duda, uno de los gestores de bases de datos popular y utilizados a la hora de desarrollar páginas web dinámicas y sitios de comercio electrónico. Se suele trabajar en combinación con PHP, y comparte con éste algunas de las características que lo convierten en una posible elección” (Gallego Vázquez, 2003). Entre ellas están:

- **Gratuito:** Se trata de software libre que puede ser utilizado sin limitación alguna.

- **Popularidad:** Son innumerables las páginas donde encontrar información, y las listas de correo donde podrán ayudar desinteresadamente con los proyectos.
- **Rapidez:** MySQL es reconocido por su alta velocidad.
- **Versatilidad:** Trabaja tanto con sistemas operativos basados en Unix como con el Sistema Operativo Windows, de Microsoft.
- **Sencillez de manejo:** Utiliza el lenguaje estándar SQL, y además el tener conocimientos de otras bases de datos ayudará enormemente y aunque no sea así, con un poco de esfuerzo puede llegar a dominarse en poco tiempo.

## PostgreSQL

“Es un sistema gestor de bases de datos objeto-relacional, distribuido bajo licencia BSD (*Berkeley Software Distribution*) y con su código fuente disponible libremente. Es el sistema gestor de bases de datos de código abierto más potente del mercado y en sus últimas versiones no tiene nada que envidiarle a otras bases de datos comerciales” (*Martínez, 2010*).

A continuación se resumen algunas de las características más importantes y soportadas por PostgreSQL:

- Integridad referencial.
- Copias de seguridad en caliente (*Online/hot backups*).
- Acceso encriptado por SSL.
- Funciones, procedimientos almacenados (*stored procedures*) en numerosos lenguajes de programación, entre otros PL/pgSQL (similar al PL/SQL de Oracle), PL/Perl, PL/Python y PL/Tcl.
- Llaves primarias (*primary keys*) y foráneas (*foreign keys*).
- *Sub-selects*.
- Consultas recursivas.

De acuerdo a las características del Sistema de Gestión Editorial para los medios de prensa cubanos v2.0 se decide utilizar PostgreSQL por ser uno de los Sistemas Gestores de Base de Datos (SGBD) de código abierto más potentes del mercado, además de ser multiplataforma. También por la necesidad de disponer de un SGBD robusto y seguro para la correcta manipulación de los datos y características propias de PostgreSQL.

## 1.5 Análisis del soporte tecnológico para el Sistema de Gestión Editorial

Actualmente la web incluye variedad en lenguajes dinámicos, desarrollo en lenguajes de presentación como HTML, CSS y Javascript. Seguidamente se analizan detenidamente algunos de los lenguajes que se utilizarán a lo largo del desarrollo del sistema.

### HTML

“Es el acrónimo de *HiperText Markup Language* que quiere decir, lenguaje de definición por marcas para hipertexto, es decir, que permite definir documentos hipertexto a base de ciertas etiquetas que marcan partes de un texto cualquiera dándoles una estructura y/o jerarquía. Para crear un documento hipertexto para la *www (World Wide Web)* con este lenguaje sólo debemos añadir las etiquetas adecuadas al texto que contiene la información. Un documento hipertexto es aquel que, además de incluir texto e ilustraciones, permite el enlace con otros documentos hipertexto que desarrollan o hacen referencia al mismo tema” (Álvarez, 2001)

### CSS

“Las hojas de estilo en cascada (*Cascading Stylesheets*) ofrecen propiedades para ampliar el lenguaje HTML en la representación visual de las páginas web. Es utilizado para definir las propiedades de formato de los diferentes elementos HTML. También permite vincular los documentos HTML con plantillas de documento, que, además de contener la información tipográfica de los elementos visuales de la página, permiten separar completamente la estructura de contenidos de su representación y presentación” (Schulz, 2008).

Algunas ventajas del lenguaje CSS son:



- Usabilidad.
- Menos código y más transparente.
- Facilita las actualizaciones.
- Accesibilidad.

## JavaScript

“Es un lenguaje de programación que permite a los desarrolladores crear acciones en sus páginas web. No requiere de compilación ya que el lenguaje funciona del lado del cliente, los navegadores son los encargados de interpretar estos códigos. Es utilizado para crear pequeños programas que luego son insertados en una página y en programas más grandes, orientados a objetos mucho más complejos y gran parte de la programación está centrada en describir objetos, escribir funciones que respondan a movimientos del mouse, aperturas, utilización de teclas, cargas de páginas, entre otros” (Pérez Valdés, 2007).

## PHP

“Es un lenguaje interpretado del lado del servidor que surge dentro de la corriente denominada código abierto (*open source*). Se caracteriza por su potencia, versatilidad, robustez y modularidad. Al igual que ocurre con tecnologías similares, los programas son integrados directamente dentro del código HTML. Fue diseñado para desarrollo de scripts orientados a web, su sintaxis es menos confusa y más estricta que la de otros lenguajes, pero sin perder la flexibilidad” (Cobo, y otros, 2005, p 36).

## 1.6 Herramientas para el desarrollo del software

Las herramientas de desarrollo son el conjunto de aplicaciones o productos informáticos utilizados por un equipo de desarrollo para cumplir con determinada tarea. A continuación, se detallan brevemente las que se utilizarán para planear, desarrollar, probar y documentar el producto a desarrollar.

### NetBeans

“Es una herramienta para que los programadores puedan escribir, compilar, depurar y ejecutar programas. Está escrito en Java, pero puede servir para cualquier otro lenguaje de programación. Es un producto libre y gratuito sin restricciones de uso” (*Oracle Coporation, 2010*).

## **Visual Paradigm para UML**

“Es una herramienta de Ingeniería de Software Asistida por Ordenador (CASE, por sus siglas en inglés), que soporta el ciclo de vida completo en el desarrollo de software: análisis y desarrollos orientados a objetos, construcción, prueba y despliegue. Permite diseñar todo tipo de diagrama de clases, código inverso, generación de código a partir de diagramas y generar documentación” (*Visual Paradigm for UML, 2013*):

## **Servidor Web Apache**

“Es el servidor web hecho por excelencia, su robustez y estabilidad hacen que cada vez millones de servidores reiteren su confianza en este programa. Es un servidor altamente configurable de diseño modular. Es muy sencillo ampliar las capacidades de éste. Actualmente existen muchos módulos para Apache que son adaptables a este, y están ahí para que los instalemos cuando los necesitemos” (*González-Vallés Saco, 2014*).

## **PgAdmin**

“Es una herramienta de código abierto para la administración de bases de datos PostgreSQL y derivados, diseñado para responder a las necesidades de la mayoría de los usuarios, desde escribir simples consultas SQL hasta desarrollar bases de datos complejas. La interfaz gráfica soporta todas las características de PostgreSQL y hace simple la administración. Está disponible en más de una docena de lenguajes y para varios sistemas operativos, incluyendo Microsoft Windows, Linux, FreeBSD, Mac OSX y Solaris” (*Reingart, 2009*).

## **1.7 Metodologías de Desarrollo**

Se utilizan para estructurar, planificar y controlar el proceso de desarrollo de un sistema de información. Existen diferentes tipos de metodologías y cada una de ellas está basada según diversas consideraciones

técnicas, organizativas y según el proyecto y el equipo. A continuación se describen algunas de las metodologías de desarrollo de software que actualmente más se utilizan.

## **Rational Unified Process (RUP):**

Es un proceso de desarrollo de software y junto con el Lenguaje Unificado de Modelado (UML por sus siglas en ingles), constituye la metodología estándar más utilizada para el análisis, implementación y documentación de sistemas orientados a objetos. “Se caracteriza por ser iterativo e incremental, estar centrado en la arquitectura y guiado por los casos de uso” (Booch, y otros, 2000, p 4).

Según Booch y colectivo de autores (2000, p 11), RUP cuenta con las cuatro fases que se describen seguidamente:

**Inicio:** Desarrolla una descripción del producto final a partir de una buena idea y se presenta el análisis de negocio para el producto. También se identifican y priorizan los riesgos más importantes, se planifica la fase de elaboración y se estima el proyecto de manera aproximada.

**Elaboración:** Especifican en detalle la mayoría de los casos de uso del sistema y se diseña la arquitectura a emplear.

**Construcción:** Se crea el producto, se añade el software terminado y la arquitectura. Al final de esta fase, el producto contiene todos los casos de uso que se definieron para el desarrollo del sistema.

**Transición:** Cubre el período durante el cual el producto se convierte en versión beta, en donde un número reducido de usuarios con experiencia prueba el producto e informa de defectos y deficiencias.

## **SXP**

Es un híbrido de metodologías compuesta por dos de ellas, *Extreme Programming* (XP) para una mayor flexibilidad en el trato con los clientes y SCRUM en la gestión de los entornos de trabajo y el equipo de desarrollo, permitiendo la mitigación de los problemas que estos pudieran ocasionar. Ofrece una estrategia tecnológica, a partir de la introducción de procedimientos ágiles que permitan actualizar los procesos de software para el mejoramiento de la actividad productiva, fomentando el desarrollo de la

creatividad, aumentando el nivel de preocupación y responsabilidad de los miembros del equipo. (Peñalver, y otros, 2010).

Consta de cuatro fases principales:

**Planificación-Definición:** Se establece la visión, se fijan las expectativas y se realiza el aseguramiento del financiamiento del proyecto.

**Desarrollo:** Donde se realiza la implementación del sistema hasta que esté listo para ser entregado.

**Entrega:** Entrega de la documentación.

**Mantenimiento:** Se realiza el soporte para el cliente.

De cada una de ellas se despliegan 7 flujos de trabajo: concepción inicial, captura de requisitos, diseño con metáforas, implementación, prueba, entrega de la documentación, soporte e investigación.

## **OpenUp**

“Es una metodología de Proceso Unificado que aplica enfoques iterativos e incrementales dentro de un ciclo de vida estructurado, basada en RUP (*Rational Unified Process*), que contiene el conjunto mínimo de prácticas que ayudan a un equipo de desarrollo de software a realizar un producto de alta calidad, de una forma eficiente” (Hernández Barrios, 2014).

Esta metodología consta de cuatro fases:

**Fase de Inicio:** Esboza las arquitecturas candidatas, identifica los riesgos, prepara un plan de proyecto y estimación de costes.

**Fase de Elaboración:** Realización de tareas de análisis del dominio y definición de la arquitectura del sistema, se elabora el plan de proyecto, se establecen los requisitos (o algunos) y la arquitectura estable del sistema y al finalizar esta fase se debe tener una definición clara y precisa de los casos de uso y todos sus actores.

**Fase de Construcción:** Es la fase más larga del proyecto, el sistema es construido en base a lo especificado en la fase de elaboración, las características del sistema se implementan en una serie de iteraciones cortas y limitadas en el tiempo, el resultado de cada iteración es una versión ejecutable del software.

**Fase de Transición:** En esta el sistema es desplegado para los usuarios finales, la retroalimentación recibida permite incorporar refinamientos al sistema en las sucesivas iteraciones, esta iteración también cubre el entrenamiento de los usuarios para la utilización del sistema.

Después de analizadas las metodologías descritas anteriormente se concluye que, SXP será la utilizada por los desarrolladores, teniendo en cuenta que el equipo de desarrollo es pequeño, además de que es una metodología ágil, tiene un enfoque centrado al cliente, está especialmente indicada para proyectos con rápido cambio de requisitos o requisitos imprecisos donde se exige una alta flexibilidad y con iteraciones cortas.

## **1.8 Conclusiones parciales**

Hasta este momento se ha podido apreciar algunos de los Sistemas de Gestión Editoriales que existen tanto en Cuba como en el mundo y sus disímiles características. Además de la justificación de las diferentes herramientas, metodologías y lenguajes que se utilizarán en el desarrollo del sistema, en donde se resume que: para el desarrollo de la propuesta de solución serán utilizados los lenguajes PHP 5.3.8, HTML 5, CSS 3 y JavaScript 1.8.

La implementación será soportada por el CMS Drupal en su versión 7.36, el cual es altamente recomendado para su utilización en instituciones periodística, además de que los sistemas desarrollados anteriormente por la universidad fueron implementados en Drupal y que el equipo de desarrollo posee experiencia en la utilización de este CMS.

## **Capítulo 2: Análisis y Diseño del Sistema de Gestión Editorial para los medios de prensa cubanos v2.0**

El objetivo de este capítulo es efectuar el análisis y diseño de la solución según los requisitos definidos para una mejor comprensión y que a su vez ayuden a lograr estructurar la arquitectura del sistema. El mismo contará con varios artefactos que tienen como objetivos mostrar el diseño del sistema propuesto como solución.

### **2.1 Descripción de la propuesta de solución**

El sistema propuesto, es configurable, modular y flexible a la estructura de diferentes medios de prensa. Pretende ser un sistema altamente confiable para el proceso editorial de estos medios, pudiéndose convertir en la herramienta preferida por el personal encargado de la gestión editorial.

Éste sistema permite al personal de cada medio, una vez autenticado dependiendo del rol que posean, acceder a un conjunto de opciones que le admitirán realizar las tareas correspondientes a su rol en el proceso editorial. De esta forma se podrán gestionar contenidos editoriales tales como: material periodístico, efemérides, cartelera de TV, fuente, entre otros; así los usuarios que gestionarán dichos contenidos.

Otra de las opciones que brinda el sistema es que una vez creado un contenido será rectificado por el Corrector. El Editor supervisará en cada fase que se encuentra el contenido y podrá seleccionar aquellos que deberán ser traducidos. De la traducción se encargarán el Jefe de Traducción y el Traductor. El Especialista de Atención al Lector se encargará de recibir y responder todas las cartas llegadas a la institución y el jefe de este departamento supervisará este proceso y seleccionará las que puedan ser publicadas. El administrador gestionará todo el sistema y podrá crear nuevos usuarios, asignarlos a diferentes roles, así como cambiar los permisos de estos.

Para la interfaz del sistema se propone la modificación de una plantilla de Drupal. Esta plantilla presenta una interfaz sencilla. Las modificaciones se realizarán siguiendo una de las estrategias marcarias de la universidad, en este caso, se selecciona Xilema que es la utilizada por el Centro de Ideoinformática. A continuación se muestra la propuesta de interfaz principal del sistema.

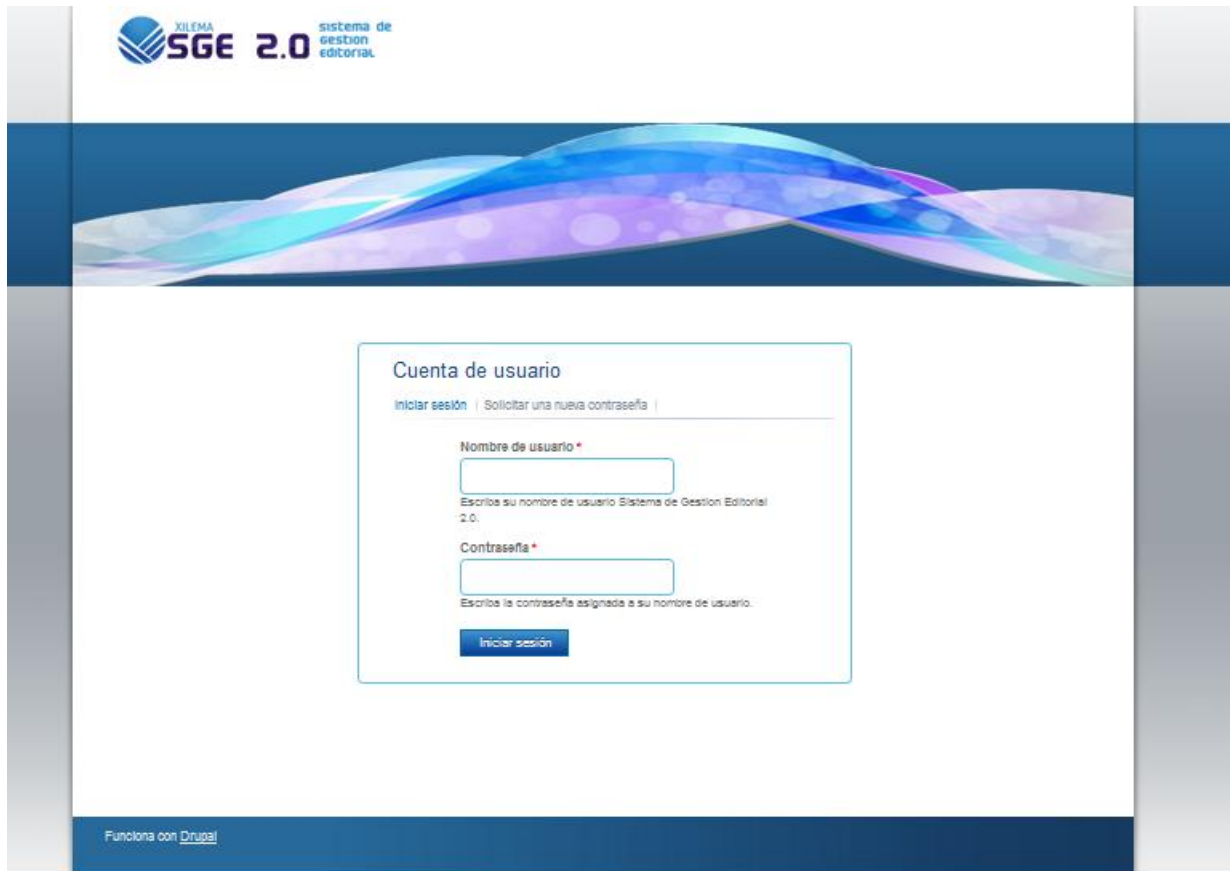


Imagen 1. Propuesta de interfaz del sistema.

## 2.2 Modelo del Dominio

El modelo del dominio muestra los tipos de objetos más importantes en el sistema y puede ser el punto de partida para el diseño de este. Es obtenido de la especificación y/o mediante la entrevista con los expertos en el tema (Fowler, 2003). Este modelo permite una mejor comprensión del sistema a desarrollar y del proceso editorial al que debe responder.

### 2.2.1 Descripción de Clases del Modelo del Dominio.

**Sistema:** Sistema de Gestión Editorial.

**Material Periodístico:** Principal contenido del sistema y del medio de prensa. Puede pertenecer a distintas categorías (noticia, reportaje, etc.).

**Efemérides:** Contiene los principales acontecimientos ocurridos en determinada fecha.

**Estado del tiempo:** Constituye una descripción del estado meteorológico.

**Canal de TV:** Canal de televisión.

**Cartelera de TV:** Lista de horarios y programas para determinado canal y fecha.

**Opinión Grafica:** Contenido donde se narra o comenta algún suceso mediante imágenes o caricaturas.

**Imagen:** Imagen o fotografía que acompañará a los contenidos.

**Correspondencias:** Cartas o mensajes recibidas por el medio de prensa que pueden ser respondidas y publicadas.

**Traducción:** Representa una traducción de un contenido a determinado idioma.

**Autor:** Persona que crea el material periodístico.

**Fuente:** Origen del material periodístico.

**Categoría:** Categoría a la que corresponde la correspondencia o el material periodístico.

**Soporte Digital:** Se refiere al destino final en una publicación en la web del material periodístico.

**Soporte Impreso:** Se refiere al destino final en una publicación impresa del material periodístico.

**Número de Edición:** Número de la publicación impresa.

**Página de número de edición:** Páginas que forman parte de un número de edición.

**Administrador:** Es la persona que puede interactuar con la aplicación libremente, administra toda la información y asigna roles y permisos de acceso a la aplicación.



**Editor:** Encargado de la aprobación de las informaciones publicadas y con la posibilidad de gestionar todos los tipos de publicaciones y términos del sistema.

**Periodista:** Personal del periódico que redacta o edita los materiales periodísticos.

**Fotógrafo:** Personal del periódico que toma las fotos donde ocurren los hechos informativos. Es el encargado de seleccionar las mejores fotografías, editarlas y ponerlas a disposición del Periodista.

**Especialista de Atención al Lector:** Personal del periódico que se encarga de recibir y dar respuestas a las cartas llegadas a la institución.

**Jefe de Atención al Lector:** Persona que autoriza que cartas saldrán publicadas.

**Corrector:** Encargado de corregir el material periodístico con el fin de darle claridad, concisión y armonía, agregando valor al texto, y haciéndolo entendible para el destinatario siguiendo las normas de redacción del periódico.

**Traductor:** Encargado de llevar a cabo la traducción y corrección de los materiales periodísticos en los idiomas: inglés, francés, italiano, portugués y alemán.

**Jefe de Traducción:** Encargado de controlar la traducción de los materiales.

**Diseñador:** Encargado de diseñar y gestionar la representación de los materiales periodísticos en las diferentes publicaciones.

## **2.2.2 Diagrama de Clases del Modelo de Dominio**

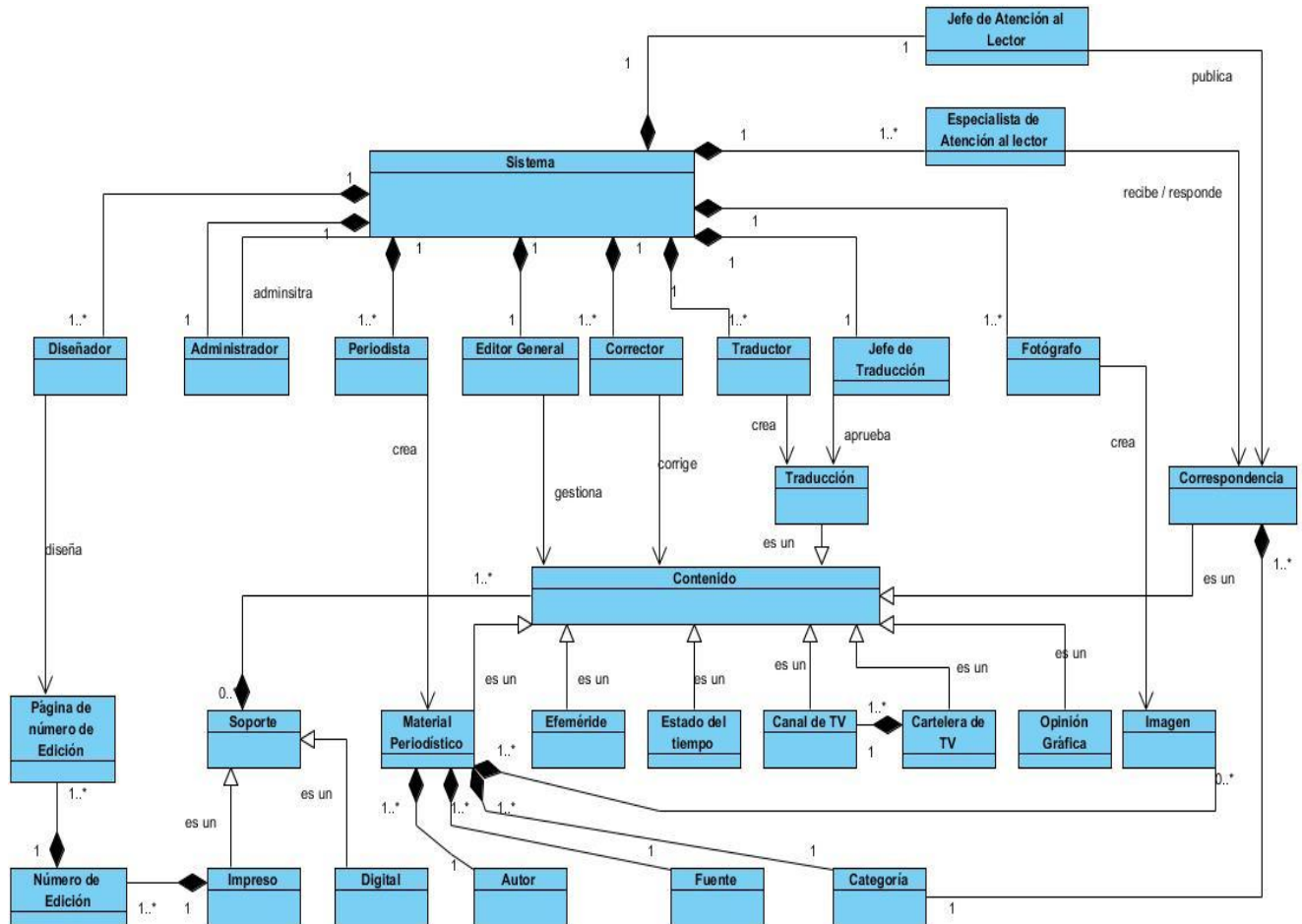


Diagrama 1. Diagrama de clases del Modelo de Dominio

## 2.3 Captura de Requisitos del Sistema de Gestión Editorial para los medios de prensa cubanos v2.0

El proceso de captura de requisitos es una etapa de suma importancia dentro del proceso de desarrollo de software, ya que marca el punto de partida para las siguientes fases del mismo. Este proceso se encarga de descubrir y analizar las necesidades del usuario del sistema a implementar y a su vez se detallan los roles que interactúan con el sistema.

### 2.3.1 Lista de Reserva del producto (LRP)

La Lista de Reserva del Producto es uno de los artefactos generados en la fase de captura de requisitos, se conforma por una lista que define los requisitos tanto funcionales como no funcionales que estarán en

el sistema a desarrollar (*Peñalver, y otros, 2010*) . Seguidamente se muestran algunos de los requisitos principales correspondiente al desarrollo del Sistema de Gestión Editorial para los medios de prensa cubanos v2.0. La LRP se encuentra en el [Anexo 1](#).

| Prioridad | No. | Descripción                            |
|-----------|-----|--|
| Alta      |     |  |
|           | 1   | Crear Usuarios.                        |
|           | 2   | Editar Usuarios.                       |
|           | 3   | Eliminar Usuarios.                     |
|           | 4   | Autenticar usuario.                    |
|           | 5   | Crear Categoría de correspondencia.    |
|           | 6   | Editar Categoría de correspondencia.   |
|           | 7   | Eliminar Categoría de correspondencia. |
|           | 8   | Crear Canal de TV.                     |
|           | 9   | Editar Canal de TV.                    |
|           | 10  | Eliminar Canal de TV.                  |
|           | 11  | Crear Material Periodístico.           |
|           | 12  | Editar Material Periodístico.          |
|           | 13  | Eliminar Material Periodístico.        |
|           | 14  | Crear Correspondencia.                 |
|           | 15  | Editar Correspondencia.                |

|       |    |   |
|-------|----|---|
|       | 16 | Eliminar Correspondencia.   |
|       | 17 | Crear Página de Número de Edición.  |
|       | 18 | Editar Página de Número de Edición.   |
|       | 19 | Eliminar Página de Número de Edición.   |
|       | 20 | Crear Número de Edición.  |
|       | 21 | Editar Número de Edición.   |
|       | 22 | Eliminar Número de Edición.   |
|       | 23 | Crear Traducción.   |
|       | 24 | Editar Traducción.  |
|       | 25 | Eliminar Traducción.  |
|       | 26 | Listar Materiales periodísticos por filtros avanzados.                              |
|       | 27 | Listar Correspondencias por filtros avanzados.                                      |
|       | 28 | Listar Páginas de Números de Edición.   |
|       | 29 | Listar Números de Edición.  |
|       | 30 | Recibir contenido de correspondencia por correo electrónico.                        |
|       | 31 | Recibir contenido de material periodístico por correo electrónico.                  |
| Media |    |   |
|       | 32 | Editor de texto con interfaz WYSIWYG que soporte las marcas de texto fundamentales. |

|      |    |   |
|------|----|---|
|      | 33 | Importar elementos desde un archivo CSV.  |
| Baja |    |   |
|      | 34 | Exportar contenido a HTML y PDF.  |
|      | 35 | Mostrar reporte de productividad de un usuario en un intervalo de tiempo determinado. |

**Tabla 1. Listado de Requisitos Funcionales.**

| Prioridad  | No. | Descripción  |
|--|-----|--|
| <b>Requisitos No Funcionales (RNF) de Apariencia o interfaz externa.</b> |     |  |
|  | 36  | Las páginas generadas deben ser compatibles con XHTML 1.0 y CSS 2.0.   |
| <b>Requisitos No Funcionales (RNF) de Seguridad.</b>                     |     |  |
|  | 37  | El sistema debe permitir usuarios con diferentes niveles de acceso.  |
|  | 38  | Se asignarán los permisos de acceso y de edición en dependencia del rol que desempeñe cada usuario del sistema.            |
|  | 39  | La información manejada por el sistema estará protegida de acceso no autorizado y divulgación.                             |
|  | 40  | La información manejada por el sistema será objeto de cuidados a protección contra la corrupción y estados inconsistentes. |
|  | 41  | Establecer un tiempo de inactividad para cerrar automáticamente las sesiones abiertas inutilizadas.                        |
|  | 42  | Solo podrá existir una sesión abierta para cada usuario en determinado tiempo.   |
| <b>Requisitos No Funcionales (RNF) de Rendimiento.</b>                   |     |  |
|  | 43  | Páginas lo más ligeras posibles, con imágenes optimizadas para la web.   |
| <b>Requisitos No Funcionales (RNF) de Hardware.</b>                      |     |  |

|  |    |   |
|--|----|---|
|  | 44 | Servidor de base de datos PostgreSQL con un Quad Core a 3.0GHz, de 2Gb de RAM y una capacidad de almacenamiento de 500 GB.    |
|  | 45 | Servidor aplicaciones <i>web</i> Apache con un Quad Core a 3.0GHz, de 2Gb de RAM y una capacidad de almacenamiento de 500 GB. |

**Tabla 2. Lista Reserva de Productos (LRP).**

### 2.3.2 Historias de usuarios y prototipos de interfaz

Son utilizadas para describir los requisitos de un software, estas definen lo que se debe construir en un proyecto de software y tienen a su vez una prioridad asociada, definida por el cliente para indicar las más importantes para el resultado final (*Beas, 2011*). A continuación se muestran algunas de las historias de usuarios del sistema a implementar.

| Historia de Usuario   |  |
|---|--|
| <b>Número:</b> 10   | <b>Nombre Historia de Usuario:</b> Gestionar Material Periodístico |
| <b>Modificación de Historia de Usuario Número:</b> 1  |  |
| <b>Usuario:</b> Lázaro R Falcón.  | <b>Iteración Asignada:</b> Primera Iteración                       |
| <b>Prioridad en Negocio:</b> Muy Alta   | <b>Puntos Estimados:</b> 3 días                                    |
| <b>Riesgo en Desarrollo:</b> ---  | <b>Puntos Reales:</b> 4 días                                       |
| <b>Descripción:</b> Disponible para el periodista y el editor. Se crea, edita y elimina un Material Periodístico.   |  |
| <b>Para crear un Material Periodístico:</b>   |  |
| <p>En el menú gestión editorial, se escoge la opción “crear material periodístico” y se procede a llenar los campos que son requeridos: título, estado, cuerpo de texto, fuente, palabras clave, temática fundamental, categoría del material, por, y los campos opcionales: localización, fecha del suceso, personas e imágenes que forman parte del material y que sirven como apoyo visual, luego se oprime el botón guardar. El sistema muestra un mensaje indicando que el material periodístico ha sido creado.</p> |  |

## **Para editar un Material Periodístico:**

En el menú gestión editorial, se escoge la opción listar material periodístico y se busca el material periodístico que se desea editar, se abre y se selecciona la pestaña editar o se hace clic directamente en la opción editar que se muestra para cada material. Después se modifican los datos, luego se presiona el botón guardar. El sistema muestra un mensaje indicando que el material periodístico ha sido actualizado correctamente.

## **Para eliminar un Material Periodístico:**

En el menú gestión editorial, se escoge la opción listar material periodístico y se busca el material periodístico que se desea eliminar, se selecciona editar y se hace clic en el botón eliminar o se hace clic directamente en la opción eliminar que se muestra para cada material. Se muestra un mensaje de confirmación y se presiona nuevamente el botón eliminar. El sistema muestra un mensaje indicando que el material periodístico ha sido borrado correctamente.

## **Observaciones:**

- Todos los campos que tengan un símbolo (\*) son obligatorios.
- El campo Fecha deberá presentar el formato correcto (Mes/Día/Año).
- La imagen subida no podrá sobrepasar 2 MB y solo se admiten archivos con las extensiones png gif jpg jpeg.
- Al seleccionar Fotorreportaje como categoría del material se deberá seleccionar una galería fotográfica.

## **Prototipos de interfaz:**

**Crear un Material Periodístico:** ver [Anexo 2](#)

**Editar un Material Periodístico:** ver [Anexo 3](#)

Eliminar un Material Periodístico: ver [Anexo 4](#)

## Historia de Usuario

Número: 14      Nombre Historia de Usuario: Gestionar Correspondencia

### Modificación de Historia de Usuario Número: 1

Usuario: Ively Viera Fajardo      Iteración Asignada: Primera Iteración

Prioridad en Negocio: Muy alta      Puntos Estimados: 3 días

Riesgo en Desarrollo: ---      Puntos Reales: 4 días

**Descripción:** Disponible para el jefe de atención al lector y el especialista de atención al lector. Se crea, edita y elimina la Correspondencia.

#### Crear Correspondencia:

En el menú gestión editorial, el corrector o especialista de atención al lector accede al vínculo crear contenido → correspondencia. Luego llena los campos obligatorios: título, cuerpo del texto, fecha, temática fundamental, categoría de correspondencia y remitente. Al presionar el botón guardar el sistema mostrará un mensaje notificando que esta ha sido creada.

#### Editar Correspondencia:

En el menú gestión editorial, se escoge la opción listar correspondencias y se busca la correspondencia que se desea editar, se abre y se selecciona la pestaña editar o se hace clic directamente en la opción editar que se muestra para cada correspondencia. Después se modifican los datos, luego se presiona el botón guardar. El sistema muestra un mensaje indicando que la correspondencia ha sido actualizada correctamente.

#### Eliminar Correspondencia :



En el menú gestión editorial, se escoge la opción listar correspondencias y se busca la correspondencia que se desea eliminar, se selecciona editar y se hace clic en el botón eliminar o se hace clic directamente en la opción eliminar que se muestra para cada correspondencia. Se muestra un mensaje de confirmación y se presiona nuevamente el botón eliminar. El sistema muestra un mensaje indicando que la correspondencia ha sido borrada correctamente.

#### Observaciones:

- Todos los campos que tengan un símbolo (\*) son obligatorios.
- El campo Fecha deberá presentar el formato correcto (Mes/Día/Año).
- Solo se podrá añadir un remitente.

#### Prototipos de interfaz:

Crear una Correspondencia: ver [Anexo 5](#)

Editar una Correspondencia: ver [Anexo 6](#)

Eliminar una Correspondencia: ver Anexo [Anexo 7](#)

#### Historia de Usuario

|            |   |
|------------|---|
| Número: 22 | <b>Nombre Historia de Usuario:</b> Listar Materiales periodísticos por filtros avanzados. |
|------------|---|

#### Modificación de Historia de Usuario Número: 1

|                                   |  |
|-----------------------------------|--|
| <b>Usuario:</b> Lázaro R. Falcón  | <b>Iteración Asignada:</b> Primera Iteración |
| <b>Prioridad en Negocio:</b> Alta | <b>Tiempo Estimados:</b> 2 días              |
| <b>Riesgo en Desarrollo:</b> ---  | <b>Tiempo Reales:</b> 3 días                 |

**Descripción:** Disponible para el periodista, corrector, editor y administrador.

En el menú gestión editorial, se selecciona listar materiales periodísticos. Se muestran los filtros título, categoría, fecha del suceso, fecha del suceso, estado actual y por. Se llena alguno de estos filtros y se oprime el botón buscar y el sistema muestra una lista de los materiales que cumplan con estos criterios. Si no se selecciona ningún filtro se mostrarán todos los materiales existentes.

**Observaciones:**

- Todos los campos que tengan un símbolo (\*) son obligatorios.
- El campo Fecha deberá presentar el formato correcto (Año-Mes-Día).

**Prototipo de interfaz:** ver [Anexo 8](#)

| Historia de Usuario   |  |
|---|--|
| <b>Número:</b> 4  | <b>Nombre Historia de Usuario:</b> Gestionar Palabra Clave |
| <b>Modificación de Historia de Usuario Número:</b> 1  |  |
| <b>Usuario:</b> Ively Viera Fajardo   | <b>Iteración Asignada:</b> Primera Iteración FD            |
| <b>Prioridad en Negocio:</b> Muy alta   | <b>Puntos Estimados:</b> 3 días                            |
| <b>Riesgo en Desarrollo:</b> ---  | <b>Puntos Reales:</b> 4 días                               |
| <p><b>Descripción:</b> Disponible para el editor, asistente de redacción, jefe de atención al lector, jefe de traducción y periodista. Se crea, edita y elimina una palabra clave.</p> <p><b>Crear palabra clave:</b></p> <p>En el menú gestión editorial, se selecciona añadir términos, se selecciona la terminología a la que se le va añadir el término y se llena el campo valor(es). Pueden ser añadidos más de un término usando</p> |  |

una “,”. Se oprime el botón guardar. El sistema muestra un mensaje indicando que la(s) palabra(s) clave ha(n) sido creada(s).

### **Editar palabra clave:**

En el menú gestión editorial, se selecciona listar términos. El sistema permite filtrar las palabras seleccionando el nombre y el vocabulario presionando el botón buscar. Se selecciona la opción editar de la palabra clave que se desea modificar, se procede a cambiar los campos y se presiona el botón guardar. Realizadas estas acciones el sistema muestra un mensaje indicando que los cambios han sido guardados.

### **Eliminar palabra clave:**

En el menú gestión editorial, se selecciona listar términos. El sistema permite filtrar las palabras seleccionando el nombre y el vocabulario presionando el botón buscar. Se selecciona la opción editar de la palabra clave que se desea eliminar, se presiona el botón eliminar, el sistema muestra un mensaje de confirmación se presiona nuevamente el botón eliminar o cancelar para revertir el proceso. El sistema muestra un mensaje indicando que la palabra clave ha sido eliminada.

### **Observaciones:**

Todos los campos que tengan el símbolo (\*) son obligatorios.

### **Prototipos de interfaz:**

**Crear una palabra clave:** ver [Anexo 9](#)

**Editar una palabra clave:** ver [Anexo 10](#)

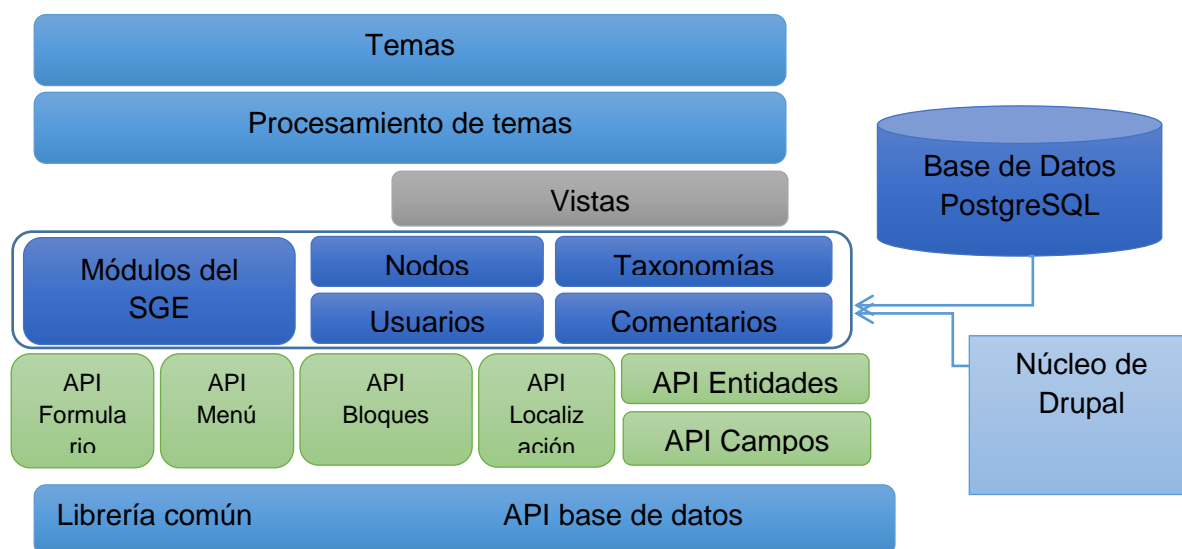
**Eliminar una palabra clave:** ver [Anexo 11](#)

## 2.4 Patrón Arquitectónico

Al ser utilizado el CMS Drupal para el desarrollo del portal, la arquitectura y los patrones son heredados del mismo. Una de las opiniones más generalizadas es que utiliza para su funcionamiento el patrón arquitectónico N-Capas, que viene dada por dos elementos fundamentales. El primero es la posibilidad de separar los diferentes aspectos del desarrollo permitiendo intercambiar porciones de la aplicación sin modificarla completamente. El otro elemento es la adaptabilidad de Drupal mediante la adición de módulos que permiten modificar partes de la aplicación, sin tener que realizar cambios en el resto de esta.

“Drupal está estructurado por temas de interfaces de interacción para con los usuarios, los cuales se pueden descargar de Internet o simplemente crearlos como plantillas en PHP, HTML y CSS. Posee una capa de abstracción de base de datos implementada y soportada para PostgreSQL, aunque puede añadir soporte para varias bases de datos. Los módulos que contienen algoritmos en PHP que proporcionan el funcionamiento del sistema, actúan como páginas servidoras y separan la interfaz gráfica de la información” (Piñero González, y otros, 2013, p 60).

Según la bibliografía consultada el CMS Drupal sigue un modelo de programación estructurada y hace uso de un sistema de bases de datos relacional. Aunque existen numerosas opiniones sobre una arquitectura única de Drupal, se pueden resumir en la siguiente imagen.



## Imagen 2. Arquitectura del CMS Drupal (*VanDyk, 2011*).

Los elementos que componen la arquitectura de Drupal 7 son los siguientes (*VanDyk, 2011*):

- **Núcleo (*core*).** Es posible acceder al núcleo y usar sus funciones a través de la Interfaz de Programación de Aplicaciones (API) de programación de Drupal.
- **Módulos (*modules*).** Los módulos aportan funcionalidades adicionales al núcleo de Drupal.
- **Área de Administración (*admin*).** Toda la administración del sitio se lleva a cabo a través del menú de administración.
- **Nodos y tipos de contenido (*nodes & types content*).** Un tipo de contenido es una plantilla a partir de la cual se crearán los contenidos. Los contenidos son las páginas finales que se mostrarán en el sitio.
- **Entidades y campos (*entities & fields*).** Las entidades son elementos a los que se les puede añadir campos. Algunas entidades en Drupal son los usuarios, los nodos, los términos de taxonomía y los comentarios.
- **Menús (*menus*).** Los menús nos ayudan a tener organizados los nodos publicados.
- **Bloques (*blocks*).** Los bloques son contenidos dinámicos que se pueden situar en las regiones del sitio.
- **Temas (*themes*).** Los temas son diseños específicos para el sitio web.
- **Usuarios, roles y permisos (*Users, roles, permissions*).** Un rol es un conjunto de permisos, y cada usuario puede tener asignados diferentes roles.
- **Taxonomía (*taxonomy*).** La taxonomía permite la clasificación de los contenidos del sitio. Está compuesta por vocabularios y términos (categorías y etiquetas).
- **Interfaz de Programación de Aplicaciones (API por sus siglas en inglés).** Librería de funciones para acceder a determinados datos del sistema de una forma abstracta.

- **Estructura de archivos.** Contiene los archivos: */sites* (Datos de configuración, módulos y temas adicionales), */modules* (Módulos del núcleo), */themes* (Temas del sitio), */includes* (Librerías de las funciones comunes), */misc* (Almacena los ficheros JavaScript y varios iconos/imágenes disponibles para usarlos en la instalación de Drupal), */profiles* (Perfiles de instalación) y */scripts* (Contiene *scripts*).

## 2.4.1 Patrones de diseño

“Un patrón de diseño es una solución a un problema que se usa repetidamente en contextos similares con algunas variantes en la implementación” (*Hassan, 2012*). El Sistema de Gestión Editorial para los medios de prensa cubanos v2.0 está desarrollado en Drupal en el cual se pueden identificar los siguientes patrones de diseño utilizados en este sistema.

### **Singleton (Instancia única)**

“La esencia de este patrón en la Programación Orientada a Objeto (POO por sus siglas en inglés) consiste en tener una sola instancia de un objeto disponible para toda la aplicación que la contiene. Provee una instancia global permitiendo que otros objetos accedan a esta única instancia. En Drupal se puede identificar este patrón si se piensan los módulos como objetos, los cuáles pueden ser pensados como una clase con una única instancia. En general lo que diferencia un módulo en Drupal de otro es el conjunto de funciones que este contiene, garantizando así la existencia de una única instancia para un módulo y la creación de un mecanismo de acceso global único a dicho módulo” (*Mezquía Marimón, 2013, p 38*).

### **Decorator (Decorador)**

“Consiste en añadir dinámicamente una funcionalidad a un objeto, esto permite no tener que crear varias clases que hereden de una primera incorporando la nueva funcionalidad, sino crear otras clases que la implementen y se asocien a la primera. En Drupal se puede identificar este patrón si se piensa a un nodo como un objeto y haciendo uso de diversos *hook* y funciones, entre ellos *hook\_node\_load ()*, *hook\_node\_view ()* y *node\_invoke ()* los cuáles, usados en los módulos “Efemérides” y “Cartelera de TV”, permiten a los módulos acciones para manipular el comportamiento de los nodos sin la necesidad de la creación de subclases o subnodos” (*Mezquía Marimón, 2013, p 39*).

## **Observer (observador)**

“Permite que los módulos se registren como observadores de los objetos de Drupal, por ejemplo, cuando se realiza una modificación a un vocabulario en el sistema de la taxonomía de Drupal, como `hook_taxonomy_vocabulary_update ()` se llama en todos los módulos que lo implementan. Al implementar este patrón, los módulos se han registrado como observadores del objeto vocabulario” (*Drupal Community, 2010*).

## **Bridge (puente)**

“La capa de abstracción de base de datos de Drupal se implementa de una manera similar a la del patrón de diseño *Bridge*. Los módulos deben ser escritos de una manera que es independiente del sistema de base de datos que se utiliza, y la capa de abstracción que lo prevé. Nuevas capas de base de datos pueden ser escritos que se ajustan a la API (*Application Programming Interface*) definida por el puente, añadiendo soporte para los sistemas de bases de datos adicionales sin la necesidad de modificar el código del módulo” (*Drupal Community, 2010*).

## **Command (comandos)**

“Muchos de los *hooks* de Drupal utilizan el patrón de *command* para reducir el número de funciones que son necesarias para poner en práctica, pasando la operación como un parámetro junto con los argumentos. De hecho, el propio sistema de *hooks* utiliza este modelo, a fin de que los módulos no tengan que definir cada *hooks*, sino sólo los que vayan a ser implementados” (*Drupal Community, 2010*).

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

“Este patrón en la POO proporciona a más de un objeto la capacidad de atender una petición, para así evitar el acoplamiento con el objeto que hace la petición. Se forma entre los objetos una cadena, en la cual cada objeto o satisface la petición o la pasa al siguiente. En el sistema de menús de Drupal se puede identificar el patrón de cadena de responsabilidades. ” (*Mezquía Marimón, 2013, p 39*).

## 2.5 Modelo de Diseño

### 2.5.1 Diagrama de Clases de Diseño (DCD)

“Representa las especificaciones de las clases e interfaces de software en una aplicación. A diferencia de las clases conceptuales del Modelo del Dominio, las clases de diseño de los DCD muestran las definiciones de las clases software en lugar de los conceptos del mundo real” (Larman, y otros, 2003).

A continuación se muestran los Diagramas de Clases de Diseño de la HU Gestionar Material Periódico, los restantes diagramas los podrán encontrar en el [Anexo 13](#).

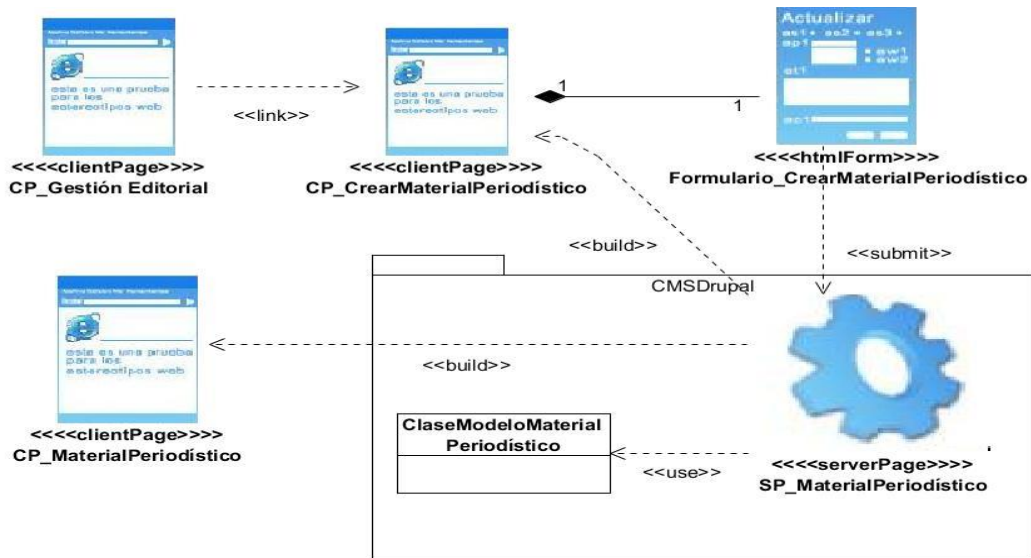
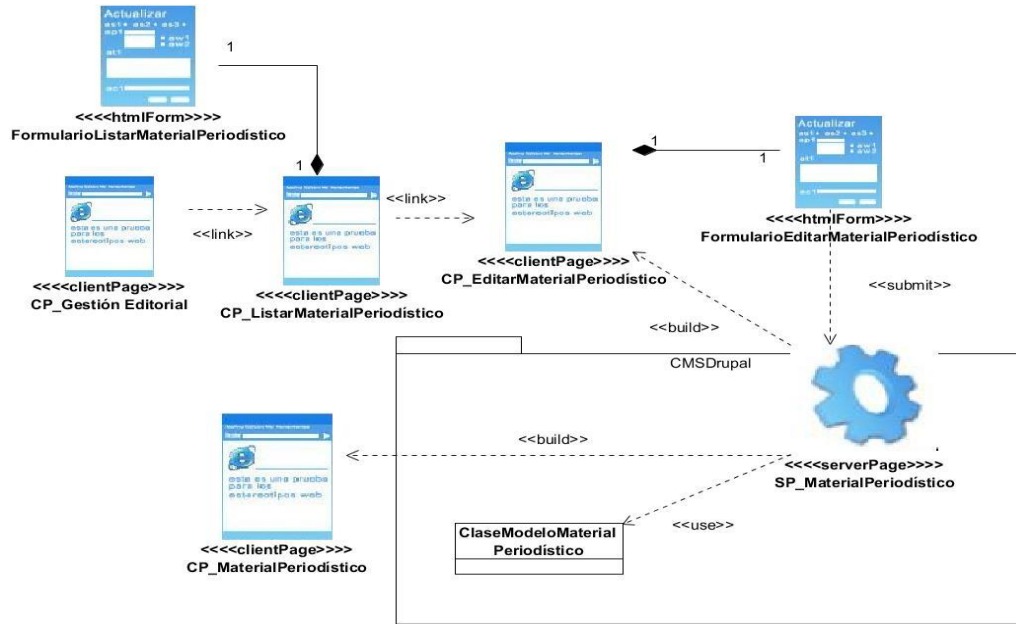
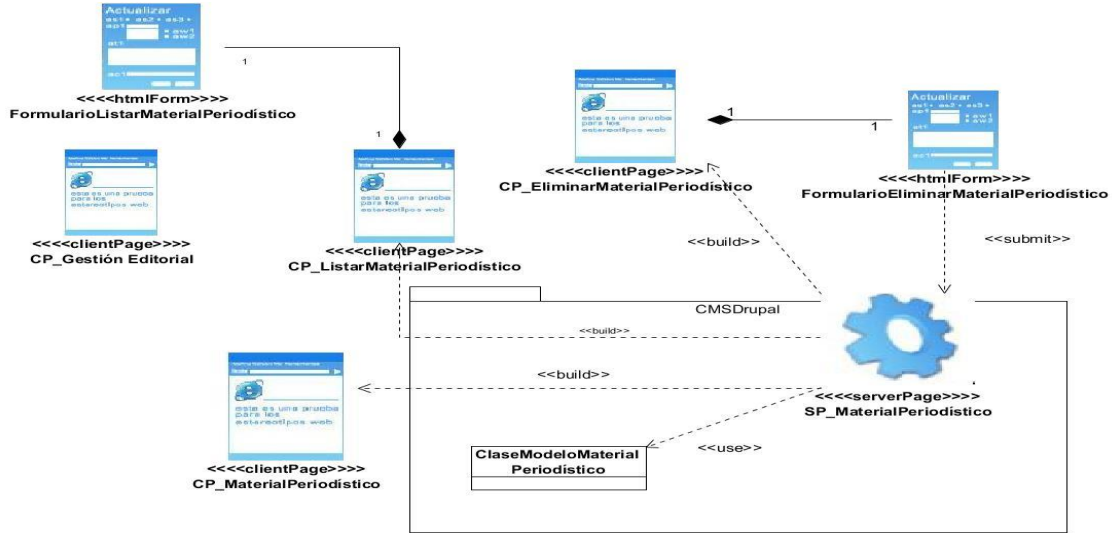


Diagrama 2. Diagrama de Clases de Diseño Escenario Crear Material Periódico

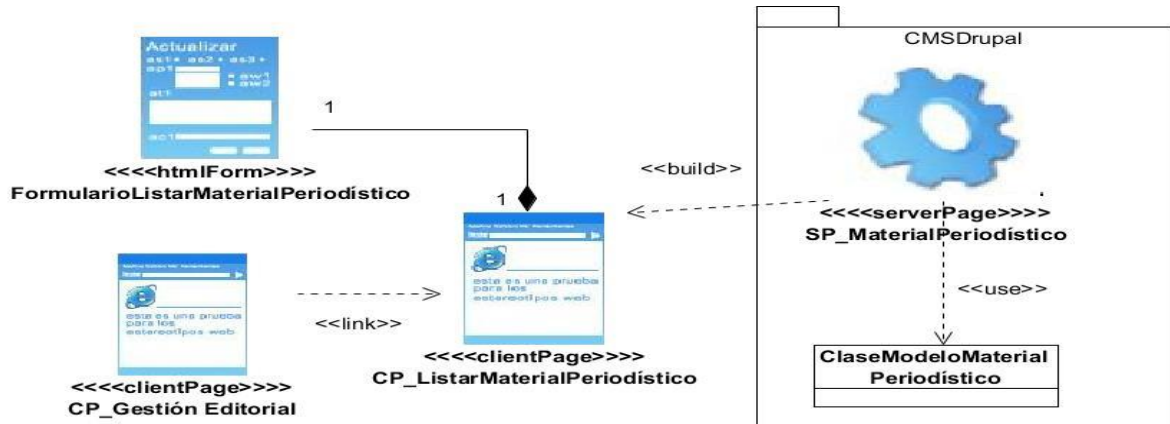




**Diagrama 3. Diagrama de Clases de Diseño Escenario Editar Material Periódístico**



**Diagrama 4. Diagrama de Clases de Diseño Escenario Eliminar Material Periódístico**



**Diagrama 5. Diagrama de Clases de Diseño Escenario Listar Material Periódico.**

## 2.6 Conclusiones parciales

En el presente capítulo se describió la propuesta de solución del sistema a partir de las deficiencias identificadas en el Sistema de Gestión Editorial para los medios de prensa cubanos. Se realizó además el modelo del dominio como parte del modelado del negocio, identificándose también los requisitos funcionales y no funcionales con que deberá cumplir el módulo, así como los diagramas de secuencia y clases de diseño con estereotipos web los cuales explican el funcionamiento del sistema.

## **Capítulo 3: Implementación y prueba del Sistema de Gestión Editorial para los medios de prensa cubanos v2.0**

En este capítulo se describen los aspectos referentes a las fases de implementación y pruebas de la propuesta de solución, se expone el diagrama de la base de datos así como el diagrama de despliegue correspondiente. Además son realizadas las pruebas ya que, durante el proceso de desarrollo del sistema, el producto debe ser verificado para ver si cumple con las expectativas del cliente y finalmente se realiza la validación del sistema propuesto.

### **3.1 Estándar de codificación**

La comunidad Drupal ha considerado necesario la introducción de una serie de estándares que faciliten la legibilidad y estructuración del desarrollo de código para Drupal. A continuación se van a ofrecer una serie de estándares para la codificación PHP en Drupal.

**Etiqueta de cierre al final del código de los archivos:** No utilizar la etiqueta de cierre `?>` al final del código de los archivos.

**Sangría de líneas:** Usar una sangría de 2 espacios en blanco, sin tabulaciones.

**Declaración de funciones:** Los nombres de las funciones deben incluir el nombre del módulo.

**Operadores:** Colocar un espacio en blanco antes y después de cada operador binario.

**Longitud de líneas superior a 80 caracteres:** Salvo las excepciones indicadas, las líneas de código no deben tener más de 80 caracteres.

**Concatenación de cadenas:** Utilizar un espacio entre el punto y las partes concatenadas.

**Nomenclatura de funciones:** Nombrar las funciones en minúscula, basándose en el nombre del módulo.

**Nomenclatura de los módulos:** No usar guiones bajos para el nombre de los módulos.

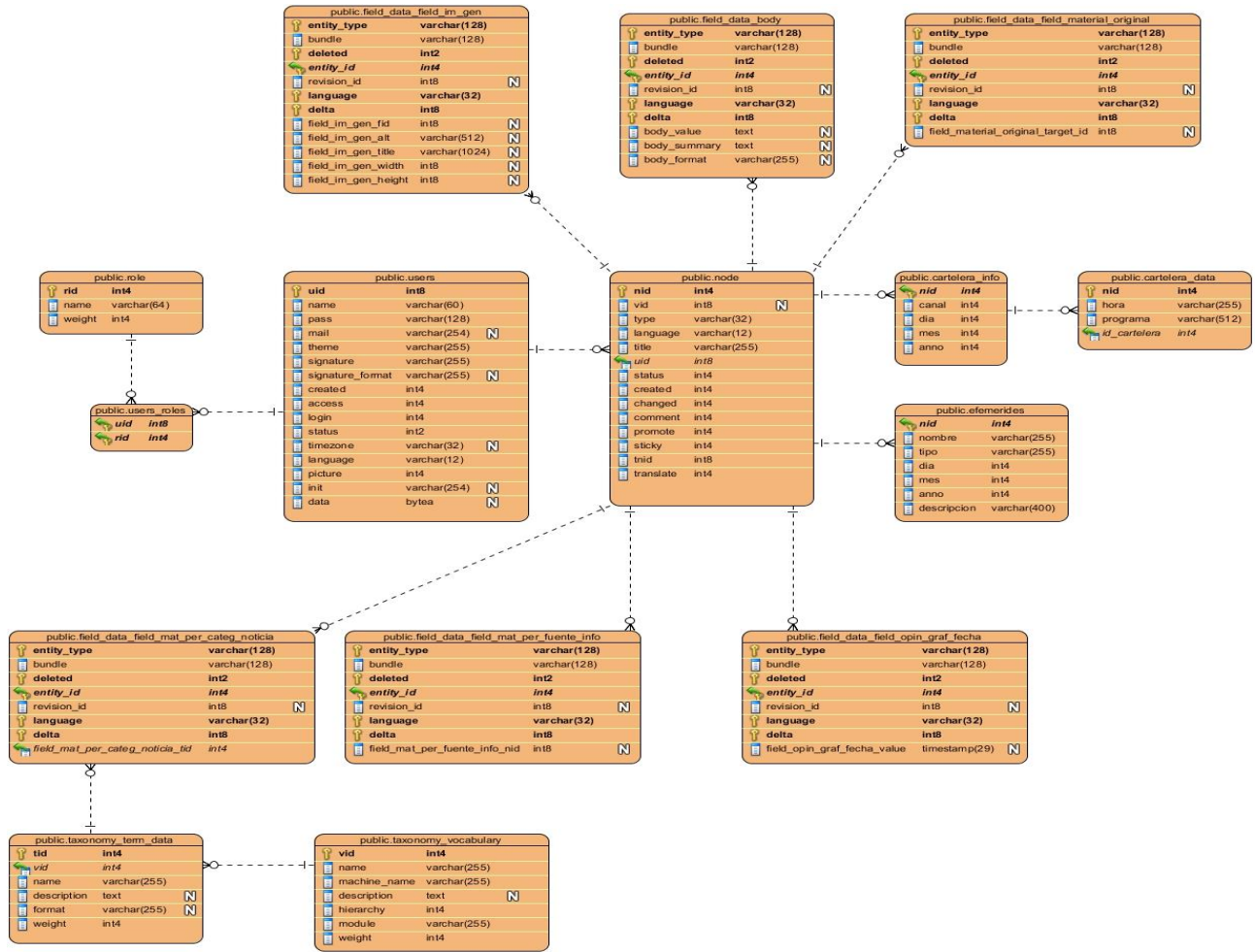
## 3.2 Plan de Entrega

| Entrega              | Descripción de la Iteración                                 | Orden de la HU a implementar   | Duración total  |
|----------------------|---|--|-----------------|
| <b>1ra Iteración</b> | Agrupar las Historias de Usuario de prioridad Alta.         | HU_1 Gestionar Usuarios.<br>HU_2 Gestionar Autor.<br>HU_3 Gestionar Fuente.<br>HU_4 Gestionar Palabra clave.<br>HU_5 Gestionar Imagen.<br>HU_6 Gestionar Mensajes para usuarios del sistema.<br>HU_7 Gestionar Categoría de correspondencia.<br>HU_8 Gestionar Respuesta de Correspondencia.<br>HU_9 Gestionar Canal de TV.<br>HU_10 Gestionar Material Periodístico.<br>HU_11 Gestionar Efemérides.<br>HU_12 Gestionar Cartelera.<br>HU_13 Gestionar Estado del Tiempo.<br>HU_14 Gestionar Correspondencia.<br>HU_15 Gestionar Opinión gráfica.<br>HU_16 Gestionar Galería Temática.<br>HU_17 Gestionar Página de Número de Edición.<br>HU_18 Gestionar Número de Edición.<br>HU_19 Gestionar Soporte Digital.<br>HU_20 Gestionar Soporte Impreso.<br>HU_21 Gestionar Traducción.<br>HU_22 Búsqueda simple.<br>HU_23 Búsqueda avanzada. | 12,7<br>semanas |
| <b>2da Iteración</b> | Agrupar las Historias de Usuario de prioridad Media y Baja. | HU_24 Almacenar nuevas revisiones.<br>HU_25 Revisión ortográfica del contenido en el idioma del mismo.<br>HU_26 Importar elementos desde un archivo CSV.<br>HU_27 Exportar contenido a HTML.<br>HU_28 Mostrar reporte de productividad de un usuario en un   | 3,5<br>semanas  |

|  |  |   |  |
|--|--|---|--|
|  |  | intervalo de tiempo determinado.<br>HU_29 Permitir seleccionar un conjunto de elementos para<br>efectuar operaciones en masa sobre ellos. |  |
|--|--|---|--|

### 3.3 Modelo de base de datos

El CMS Drupal posee un alto número de entidades manejadas por él, por lo que no se representarán todas en el modelo. En las últimas versiones de Drupal por cada campo de los tipos de contenido se generan dos tablas *field\_data\_(nombre del campo)* y *field\_revision\_(nombre del campo)*. Solo se mostrarán ejemplos de las principales tablas de este tipo ya que solo cambian en el nombre del campo. Durante el diseño de las tablas necesarias para la implementación del sistema se tuvieron en cuenta aquellas clases que debían ser persistentes en la base de datos de Drupal para cumplir con determinada funcionalidad del sistema. A continuación se muestra el diseño de la base de datos del sistema a través del modelo de datos.



**Diagrama 6. Diagrama de base de datos**

### 3.3.1 Descripción del Modelo de Datos

A continuación se describen algunas de las tablas más críticas de la base de datos, las restantes pueden ser encontradas en el [Anexo 12](#).

| Nombre: users |                                       |                            |
|---------------|---------------------------------------|----------------------------|
| Descripción   | Almacena información de los usuarios. |                            |
| Atributo      | Tipo                                  | Descripción                |
| uid           | int8                                  | Identificador del usuario. |

|                  |         |  |
|------------------|---------|--|
| name             | varchar | Nombre del usuario.                                    |
| pass             | varchar | Contraseña del usuario.                                |
| mail             | varchar | Dirección de correo electrónico del usuario.           |
| theme            | varchar | Tema por defecto del usuario.                          |
| signature        | varchar | Firma del usuario.                                     |
| signature_format | varchar | Formato de la firma del usuario.                       |
| created          | int4    | Fecha de creación del usuario.                         |
| access           | int4    | Fecha del acceso anterior del usuario.                 |
| login            | int4    | Fecha del último acceso del usuario.                   |
| status           | int2    | Estado del usuario (activo o bloqueado).               |
| timezone         | varchar | Zona horaria del usuario                               |
| language         | varchar | Lenguaje por defecto del usuario                       |
| picture          | int4    | Identificador de la tabla de las imágenes del usuario. |
| init             | varchar | Primera dirección de correo electrónico.               |
| data             | bytea   | Arreglo solo utilizado por Drupal.                     |

**Tabla 3. Descripción de la tabla *users* del Modelo de Datos.**

| Nombre: node |                                    |   |
|--------------|------------------------------------|---|
| Descripción  | Almacena información de los nodos. |   |
| Atributo     | Tipo                               | Descripción   |
| nid          | int4                               | Identificador del nodo.   |
| vid          | int8                               | Identificador de la revisión del nodo.                                  |
| type         | varchar                            | Tipo de nodo.   |
| language     | varchar                            | Idioma del nodo.  |
| title        | varchar                            | Título del nodo.  |
| uid          | int8                               | Identificador del usuario que creó del nodo.                            |
| status       | int4                               | Valor booleano que indica si el nodo está publicado o no.               |
| created      | int4                               | Fecha de creación del nodo.   |
| changed      | int4                               | Fecha de modificación del nodo.   |
| comment      | int4                               | Valor booleano que indica si están permitidos comentarios o no.         |
| promote      | int4                               | Valor booleano que indica si será publicado en la página principal.     |
| sticky       | int4                               | Valor booleano que indica si será publicado al principio de las listas. |
| tnid         | int8                               | Identificador de la traducción del nodo.                                |
| translate    | int4                               | Valor booleano que indica si la traducción debe ser actualizada.        |

**Tabla 4. Descripción de la tabla *node* del Modelo de Datos.**

## 3.4 Diagrama de Componentes

Los Diagramas de Componentes se utilizan para modelar la vista estática de un sistema. (Serrano Paret, y otros, 2012, p 53). Estos muestran la organización y dependencias entre un conjunto de componentes del sistema. A continuación se muestra el diagrama de componentes del Sistema de Gestión Editorial v2.0:

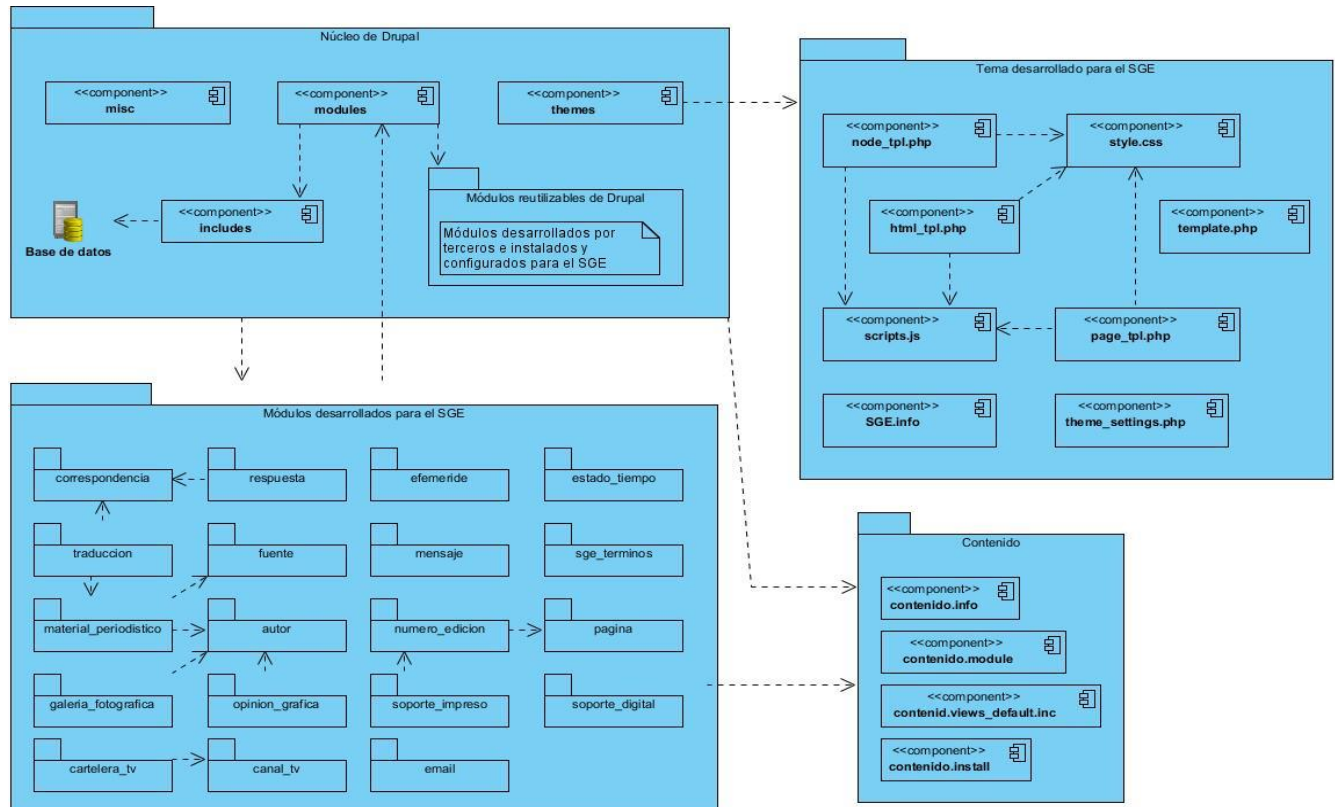


Diagrama 7. Diagrama de Componentes.

A continuación se describe cada componente representado en el diagrama anterior.

**Núcleo de Drupal:** Contiene los principales componentes básicos con los que cuenta Drupal y se crean desde que se instala este.

**Themes:** Capa de presentación y diseño de Drupal.

**Modules:** Aquí se encuentran todos los módulos de Drupal.



**Includes:** Contiene un conjunto de librerías en forma de archivos PHP con extensión .inc, que incluyen funciones comunes del sistema (ajax.inc, batch.inc, cache.inc, date.inc, form.inc).

**Misc:** Incluye archivos CSS y JavaScript para el diseño y funcionamiento de Drupal.

**Módulos reutilizables de Drupal:** Contiene los módulos desarrollados por terceros y que fueron instalados y configurados para el Sistema de Gestión Editorial para los medios de prensa cubanos (SGEMPC) v2.0.

**Tema Sistema de Gestión Editorial:** Tema diseñado y desarrollado para el SGEMPC v2.0

**Módulos desarrollados para el SGE:** Contiene los módulos desarrollados para el SGEMPC v2.0.

**Base de Datos:** Es la base de datos del sistema.

**contenido.bundle:** Archivo que permite importar tipos de contenido en Drupal.

**contenido.info:** Archivo que contiene información esencial del módulo.

**contenido.views\_default.inc:** Archivo que permite importar una vista a Drupal.

**contenido.install:** Archivo que contiene funcionalidades que se ejecutan al instalar el módulo.

**contenido.module:** Archivo principal del módulo, contiene todas las funciones del módulo.

### **3.5 Diagrama de Despliegue**

El diagrama de despliegue es un tipo de diagrama UML que se utiliza para modelar el hardware utilizado en las implementaciones de sistemas.

En la imagen siguiente se muestra el diagrama de Despliegue correspondiente al sistema a desarrollar.



**Diagrama 8. Diagrama de Despliegue**

### Descripción del Diagrama de despliegue.

**Dispositivo Cliente:** Es el dispositivo que utilizarán los usuarios para acceder al sistema. Generalmente se trata de una computadora que se comunica vía HTTPS con el servidor Web.

**Servidor Web (Apache):** Es el intermediario entre el dispositivo cliente que realiza sus peticiones y el servidor de base de datos donde se encuentran las informaciones. Este extrae los datos, realiza sus funciones y presenta la información a los dispositivos clientes. Se comunica con el servidor de base de datos mediante los protocolos TCP/IP.

**Servidor de base de datos (PostgreSQL):** Se encarga de mantener las informaciones generadas como parte de la base de datos del CMS Drupal y permite el acceso a ella mediante el Servidor web.

### 3.6 Pruebas

Luego de implementado el sistema se hace necesario realizar un conjunto de pruebas que permitan certificar la calidad de una aplicación informática. Para determinar el nivel de calidad se realizaron un conjunto de pruebas que permitieron comprobar el grado de cumplimiento de las especificaciones principales del sistema. Seguidamente se detallarán las pruebas de funcionalidad, rendimiento y seguridad realizadas a la aplicación.

#### 3.6.1 Pruebas de Aceptación

Las pruebas de aceptación son un tipo de caja negra orientadas a evaluar las distintas tareas en las que ha sido dividida una historia de usuario. Para asegurar el funcionamiento final de una determinada historia, las pruebas son creadas y usadas por los clientes para comprobar que éstas funcionen como deben. A

continuación se describe el Caso de prueba para la HU\_Gestionar Material Periodístico, los demás casos de pruebas se pueden encontrar en el [Anexo 14](#).

## Casos de pruebas para la HU\_Gestionar Material Periodístico

| Caso de Prueba de Aceptación   |   |
|--|---|
| <b>Código Caso de Prueba:</b> SGEPMC-1.1   | <b>Nombre Historia de Usuario:</b> Gestionar Material Periodístico. |
| <b>Nombre de la persona que realiza la prueba:</b> Ively Viera Fajardo   |   |
| <b>Descripción de la prueba:</b> Se crea un material periodístico en el cual los campos requeridos son: título, estado, cuerpo del texto, fuente, localización, fecha de suceso, palabras claves, personas, categoría del material, categoría, temática fundamental, por, imagen y galería fotográfica.  |   |
| <b>Condiciones de ejecución:</b> Luego de autenticado el periodista, este podrá acceder a una interfaz que le permita crear un material periodístico.  |   |
| <b>Entrada/Pasos de ejecución:</b> Una vez dentro del Sistema de Gestión Editorial para los medios de prensa cubanos v2.0 en el menú gestión editorial, se escoge la opción crear material periodístico y seguidamente se llenan todos los campos requeridos del material periodístico a crear, los campos obligatorios son: título, estado, cuerpo del texto, fuente, palabras claves, categoría del material, categoría, temática fundamental y por, los cuales se diferencian por un símbolo en color rojo (*). Se debe asegurar que el campo fecha presente el formato correcto: (Mes/Día/Año) y en caso de haber seleccionado fotorreportaje como categoría del material se deberá seleccionar una galería fotográfica. Luego se presiona el botón guardar. |   |
| <b>Resultado esperado:</b> El sistema muestra un mensaje indicando que se ha creado correctamente un material periodístico.  |   |
| <b>Evaluación de la prueba:</b> Satisfactoria.   |   |

| Caso de Prueba de Aceptación   |   |
|--|---|
| <b>Código Caso de Prueba:</b> SGEPMC-1.2   | <b>Nombre Historia de Usuario:</b> Gestionar Material Periodístico. |
| <b>Nombre de la persona que realiza la prueba:</b> Ively Viera Fajardo   |   |
| <b>Descripción de la prueba:</b> Se edita un material periodístico y en el proceso se verifica que todos los campos requeridos se llenen correctamente.  |   |
| <b>Condiciones de ejecución:</b> Luego de autenticado el periodista, este podrá acceder a una interfaz que le permita editar un material periodístico.   |   |
| <b>Entrada/Pasos de ejecución:</b> En el menú gestión editorial, se escoge la opción listar materiales periodísticos y se selecciona el material periodístico a editar. Seguidamente se modifican los campos deseados y se debe asegurar que el campo fecha presente el formato correcto: (Mes/Día/Año) y en caso de haber seleccionado fotorreportaje como categoría del material se deberá seleccionar una galería fotográfica. Luego se selecciona la opción guardar. |   |
| <b>Resultado esperado:</b> El sistema muestra un mensaje indicando que se ha editado correctamente un material periodístico.   |   |
| <b>Evaluación de la prueba:</b> Satisfactoria.   |   |

| Caso de Prueba de Aceptación   |   |
|--|---|
| <b>Código Caso de Prueba:</b> SGEPMC-1.3                               | <b>Nombre Historia de Usuario:</b> Gestionar Material Periodístico. |
| <b>Nombre de la persona que realiza la prueba:</b> Ively Viera Fajardo |   |
| <b>Descripción de la prueba:</b> Se elimina un material periodístico.  |   |

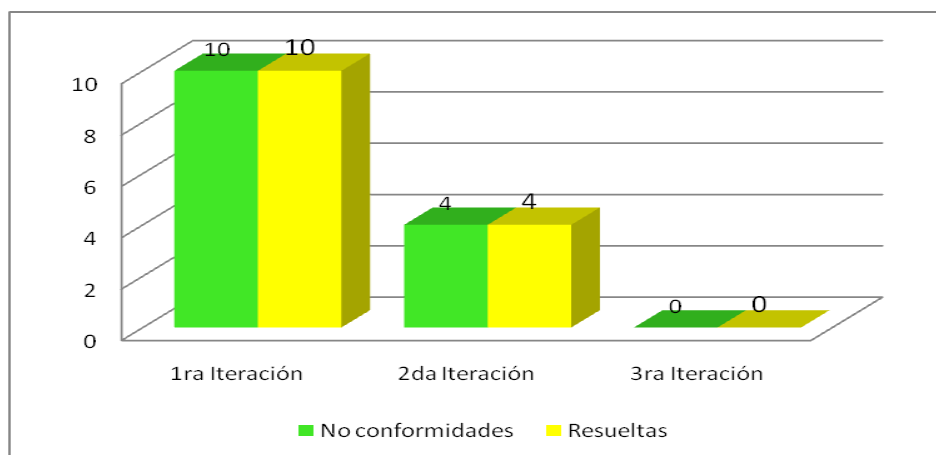
**Condiciones de ejecución:** Luego de autenticado el periodista, este podrá acceder a una interfaz que le permita eliminar un material periodístico.

**Entrada/Pasos de ejecución:** En el bloque de periodista se selecciona el material periodístico a eliminar que se encuentra en el listado de materiales creados y seguidamente el sistema muestra un mensaje para confirmar la eliminación del material periodístico.

**Resultado esperado:** El sistema muestra un mensaje indicando que se ha eliminado correctamente un material periodístico.

**Evaluación de la prueba:** Satisfactoria.

Luego de realizados un conjunto de casos de prueba se detectaron 10 no conformidades en la primera iteración que permitían que el sistema no se comportara de la forma esperada, de las cuales 6 eran errores ortográficos y 4 de validación. De estas no conformidades se solucionaron las 10. Luego de una segunda iteración se encontraron 4 no conformidades no significativas caracterizadas en errores ortográficos, de las cuales se solucionaron todas. Se ejecutó una tercera iteración obteniéndose un resultado satisfactorio para cada una de las combinaciones de datos por escenario, ya que no se encontraron no conformidades. La siguiente figura muestra estos resultados en forma de gráfica:



**Imagen 3. Comportamiento de las no conformidades por iteración.**

### **3.6.2 Pruebas de Seguridad**

La seguridad del sistema se basa en validaciones, posibles ataques informáticos, acceso a información sensible por algún personal no deseado entre otros aspectos. Para detectar y corregir problemas asociados a estos aspectos se realizan pruebas de seguridad. Estas pruebas certifican que los datos del sistema solamente sean accedidos por los actores definidos según los niveles de acceso que posean. Tienen como objetivo evaluar la confidencialidad, integridad y disponibilidad de los datos.

Para probar la seguridad del sistema se aplicaron una serie de pruebas usando el Acunetix v8.0 el cual luego de haber escaneado el sistema generó un reporte de las vulnerabilidades de seguridad más importantes divididas en alta, media, baja e informativa. En una primera iteración solo se identificaron 11 vulnerabilidades entre las que se encontraban: envío de información sensible en texto plano, enlaces rotos y el tipo de contraseña de entrada con autocompletado habilitado. De las vulnerabilidades detectadas se encontraron 6 medias, 1 baja y 4 informativas.

Se realizó una segunda iteración donde fueron corregidos todos estos fallos por lo que se concluye que el nivel de seguridad del sistema es correcto.

### **3.6.3 Pruebas de Rendimiento**

Las pruebas de rendimiento son realizadas sobre computadoras, redes, software u otros dispositivos, son utilizadas para determinar la velocidad y eficiencia de los mismos. Estas pueden ser realizadas a través de herramientas que proveen pruebas de estrés, que permiten determinar la estabilidad del sistema. (*Serrano Paret, y otros, 2012, p 53*)

Para realizar estas pruebas se usa la herramienta JMeter, la cual es diseñada para realizar Pruebas de Rendimiento sobre Aplicaciones Web, hoy en día su arquitectura ha evolucionado, ahora no sólo puede llevar a cabo pruebas en componentes típicos de Internet (HTTP), sino también puede realizar pruebas sobre Bases de Datos, scripts Perl, objetos java, servidores FTP y prácticamente cualquier medio de los que se pueden encontrar en la red (*The Apache Software Foundation, 1999*).

Las pruebas realizadas consistieron en definir dos escenarios de pruebas que consistían en simular:

**Escenario 1.** Acceso y autenticación de 20 usuarios al sistema al mismo tiempo.

**Escenario 2.** Autenticación y acceso de 20 usuarios al mismo tiempo a algunos de los listados de contenidos del sistema.

A continuación se muestran los resultados obtenidos por los dos componentes al realizar las dos pruebas diseñadas.

| No. Muestras | Media | Mediana | Línea de 90% | Min | Max   | % Error | Rendimiento | Kb/seg |
|--------------|-------|---------|--------------|-----|-------|---------|-------------|--------|
| 1060         | 314   | 36      | 176          | 11  | 12222 | 0,00%   | 29,9 seg    | 279,5  |

**Tabla 5. Resultado obtenido para el escenario 1.**

| No. Muestras | Media | Mediana | Línea de 90% | Min | Max   | % Error | Rendimiento | Kb/seg |
|--------------|-------|---------|--------------|-----|-------|---------|-------------|--------|
| 1720         | 270   | 35      | 123          | 9   | 11021 | 0,00%   | 34,4 seg    | 302,9  |

**Tabla 6. Resultado obtenido para el escenario 2.**

### 3.7 Conclusiones Parciales

Con la realización de este capítulo quedó definido el modelo de implementación del Sistema de Gestión Editorial v2.0 a través del diagrama de componentes, describiéndose cada uno de sus componentes y su influencia en la solución obtenida. Se diseñaron además, los casos de pruebas para comprobar la solución implementada, específicamente se desarrollaron pruebas de funcionalidad, seguridad, rendimiento y usabilidad.

## **Conclusiones**

Una vez concluida la presente investigación se arriban a las conclusiones siguientes:

1. A partir del estudio de diversos Sistemas de Gestión Editoriales utilizados en Cuba y el mundo, se identificaron las funcionalidades y características que no cumplen con las necesidades actuales de las instituciones periodísticas del país y que sirvieron de base para el desarrollo del Sistema de Gestión Editorial para los medios de prensa cubanos v2.0.
2. Se caracterizaron los principales lenguajes y herramientas empleados para el desarrollo de la aplicación, fundamentando en cada caso la selección.
3. La definición de los requerimientos de software en conjunto con el análisis del diseño de clases y el modelo de datos posibilitaron la definición de las funcionalidades de la aplicación informática para cumplir los objetivos planteados.
4. Como resultado de la implementación se obtuvo un sistema funcional que cumple con los requerimientos especificados.
5. La solución fue debidamente probada y las deficiencias detectadas en cada una de las iteraciones de prueba fueron corregidas en el tiempo establecido, lo que posibilitó la efectividad de la solución propuesta.

Por todo lo anteriormente expuesto se concluye que los objetivos propuestos para el presente trabajo han sido cumplidos satisfactoriamente ya que la aplicación da solución a la situación problémica que le dio origen.



## **Recomendaciones**

Aun cuando los objetivos de este trabajo fueron cumplidos, se recomienda:

- Continuar estudiando el proceso editorial en Cuba y en el mundo para incluir en la solución implementada posibles nuevas funcionalidades que puedan surgir.
- Promover el Sistema de Gestión Editorial para los medios de prensa cubanos v2.0 a las instituciones periodísticas correspondientes para su futura utilización.
- Se recomienda este trabajo como fuente de información para el desarrollo de sistemas similares o versiones posteriores de este.

## Referencias bibliográficas

1. **Barriocanal, Luis. 2009.** ¿Qué es Joomla? *edujoomla.es*. [En línea] 12 de mayo de 2009. [Citado el: 17 de octubre de 2014.] <http://www.edujoomla.es/que-es-joomla>.
2. **Álvarez, Miguel Angel. 2001.** *desarrolloweb.com*. [En línea] 1 de enero de 2001. [Citado el: 15 de noviembre de 2014.] <http://www.desarrolloweb.com/articulos/que-es-html.html>.
3. **Beas, José Manuel. 2011.** *JMBEAS*. [En línea] 23 de mayo de 2011. [Citado el: 24 de marzo de 2015.] <http://jmbeas.es/guias/historias-de-usuario/>.
4. **Booch, Grady, Rumbaugh, James y Jacobson, Ivar. 2000.** *El Proceso Unificado de Desarrollo de Software*. Madrid : s.n., 2000. pág. 4. ISBN: 84-7829-036-2.
5. **Cobo, Ángel, y otros. 2005.** *Tecnologías para el desarrollo de aplicaciones web, INTRODUCCIÓN A PHP*. 2005. pág. 36. [http://books.google.com/cu/books/about/PHP\\_y\\_MySQL.html?id=zMK3GOMOpQ4C&redir\\_esc=y](http://books.google.com/cu/books/about/PHP_y_MySQL.html?id=zMK3GOMOpQ4C&redir_esc=y).
6. **Debén González, Alejandro. Junio 2010.** *Propuesta de una representación del modelo estándar para la descripción del material periodísticos*. Facultad de Comunicación, Universidad de la Habana. La Habana : s.n., Junio 2010. pág. 1.
7. **Drupal Community. 2010.** *Drupal*. [En línea] About Drupal by Community of Drupal, Abril de 2010. [Citado el: 5 de marzo de 2015.] <https://www.drupal.org/node/547518>.
8. *Estudio de los sistemas de gestión de contenidos web. Análisis de las mejores soluciones del mercado. Centro de apoyo Tecnológico a Emprendedores, Fundación Parque Científico y Tecnológico de Albacete. 2012.* 2012, pág. 20.
9. **Experience,Soft. 2006.** *KaliNews-software editor de textos editoriales en XML para periodistas y correspondientes de la prensa-transmisión y recepción de información estructurada*. [En línea] 20 de junio de 2006. [Citado el: 20 de noviembre de 2013.] <http://peccatte.karefil.com/KaliNews/ES/Index.html>.

10. **Fowler, Martin. 2003.** *Patterns of Enterprise Application Architecture.* 2003.
11. **Gallego Vázquez, Jose Antonio. 2003.** Desarrollo Web con PHP y MySQL. Madrid : Ediciones Anaya Multimedia, 2003.
12. **González-Vallés Saco, Guillermo. 2014.** Ciberula. [En línea] 2014. [Citado el: 10 de noviembre de 2014.] [http://linux.ciberaula.com/articulo/linux\\_apache\\_intro](http://linux.ciberaula.com/articulo/linux_apache_intro).
13. **Hassan, Yusef. 2012.** Human-Computer. [En línea] 9 de noviembre de 2012. [Citado el: 20 de enero de 2015.] <http://www.human-computer.net/blog/2012/11/patrones-de-diseno-de-interaccion/>.
14. **Hernández Barrios, Carmelo José. 2014.** SlideShare. [En línea] 20 de septiembre de 2014. [Citado el: 10 de noviembre de 2014.] <http://es.slideshare.net/carmeloh2/metodologa-open-up-39321348>.
15. **Hista Internacional. 2007.** Consultoría en Metodologías de Desarrollo de Software - RUP y las mejores prácticas para el desarrollo de software. [En línea] 2007. [Citado el: 25 de noviembre de 2014.] <http://www.histaintl.com/servicios/consulting/rup.php>.
16. **Jeréz Abreu, Denia y Díaz Mendiluz, Yarelis. 2010.** *Personalización del Sistema de Gestión de Contenidos Drupal para la implementación de Revistas Científicas Electrónicas.* Universidad de las Ciencias Informáticas. La Habana : s.n., 2010. pág. 26.
17. **Jiménez-Hidalgo, Sonia, Giménez-Toledo, Elea y Salvador-Bruna, Javier. 2008.** Los sistemas de gestión editorial como medio de mejora de la calidad y la visibilidad de las revistas científicas. *El profesional de la información.* 2008, pág. 282.
18. *La noticia y la entrevista. Una aproximación a su concepto y estructura.* **Yanes Mesa, Rafael. 2002.** 9-10, 2002, pág. 251.
19. **Larman, Craig y Hall, Prentice . 2003.** El Modelo de Diseño. *UML y Patrones.* 2da. 2003.
20. *Las tecnologías de información en la actividad editorial: tendencias, contextos y perspectivas.* **Rodríguez Roche, Sulema y Leiva Ramos, Aimée. 2009.** 5, La Habana : s.n., noviembre de 2009,

Vol. 20. [http://scielo.sld.cu/scielo.php?script=sci\\_arttext&pid=S1024-94352009001100005&lng=es&nrm=iso](http://scielo.sld.cu/scielo.php?script=sci_arttext&pid=S1024-94352009001100005&lng=es&nrm=iso). ISBN 1024-9435.

21. **Macías Rodríguez, Juan y Vázquez Rodríguez, Annarelys. junio del 2007.** *Estudio de Sistemas de Gestión de Contenidos basados en lenguaje PHP*. La Habana : s.n., junio del 2007. pág. 42.
22. **Martínez Bravo, Indira. 2007.** Informática. Trabajos investigativos sobre sistemas, softwares, tecnología informática, entre otros. [En línea] septiembre de 2007. [Citado el: 16 de noviembre de 2014.] <http://indira-informatica.blogspot.com/2007/09/qu-es-un-sistema-de-gestin-de-base-de.html>.
23. **Martínez, Rafael. 2010.** PostgreSQL\_es. [En línea] 2 de octubre de 2010. [Citado el: 10 de noviembre de 2014.] [http://www.postgresql.org.es/sobre\\_postgresql](http://www.postgresql.org.es/sobre_postgresql).
24. **Mezquía Marimón, Diosbel. 2013.** *Perfil de instalación de portales intranet para el Sistema Gestor de Contenidos Drupal 7*. La Habana : s.n., 2013.
25. **Moreno Espinosa, Pastora. 2000.** *Los géneros periodísticos informativos en la actualidad internacional*. Sevilla : s.n., 2000. pág. 173. <http://grupo.us.es/grehcco/ambitos05/pastora.pdf>.
26. —. **2003.** Sala de Prensa. [En línea] 2003. [Citado el: 20 de noviembre de 2014.] <http://www.saladeprensa.org/art501.htm>.
27. **Oracle Coporation. 2010.** NetBeans. [En línea] 2010. [Citado el: 16 de noviembre de 2015.] <https://netbeans.org/community/index.html>.
28. **Peñalver, G, Meneses, A y García, A. 2010.** *SXP Metodología ágil para el desarrollo de software*. Universidad de las Ciencias Informáticas. La Habana : s.n., 2010. pág. 337.
29. **Peñalver, G, Meneses, A y García, S. 2010.** *SXP, Metodología ágil para el desarrollo de software*. Universidad de las Ciencias Informáticas. La Habana : s.n., 2010.
30. **Pérez López, Georgina. 2007.** Scribd. [En línea] 2007. [Citado el: 20 de noviembre de 2014.] <https://es.scribd.com/doc/270248/EI-Reportaje>.

31. **Pérez Valdés, Damián. 2007.** Maestros del Web. [En línea] 3 de julio de 2007. [Citado el: 15 de noviembre de 2014.] <http://www.maestrosdelweb.com/que-es-javascript/>.
32. **Piñero González, Maidelyn , Jiménez Morales, Yenier y Kubota López, Emilio . 2013.** Serie Científica. [En línea] 2013. [Citado el: 20 de enero de 2015.] <http://publicaciones.uci.cu/index.php/SC/article/view/1344/706>. ISSN: 2306-2495.
33. **Reingart, Mariano. 2009.** PostgreSQL. [En línea] 28 de diciembre de 2009. [Citado el: 15 de noviembre de 2014.] <http://www.postgresql.org/message-id/5aebd8250912271958m9533409td5bfee02364eda50@mail.gmail.com>.
34. **Robertson, J. 2002.** How to evaluate a content management system. *Step Two* . [En línea] 23 de enero de 2002. [Citado el: 12 de noviembre de 2014.] [http://www.steptwo.com.au/papers/kmc\\_evaluate/](http://www.steptwo.com.au/papers/kmc_evaluate/).
35. —. **2003.** So, what is a content management system? [En línea] 3 de junio de 2003. [Citado el: 11 de noviembre de 2014.] [http://www.steptwo.com.au/papers/kmc\\_what/](http://www.steptwo.com.au/papers/kmc_what/).
36. **Schulz, Ralph G. 2009.** *Diseño web con CSS*. Marcombo: s.n., 2009. pág. 4. ISBN: 978-84-267-1470-1.
37. **Serrano Paret, Johan y Galano Segura, Alfredo. 2012.** *Sistema de Gestión Editorial para los Medios de Prensa Cubanos*. La Habana : s.n., 2012. pág. 1.
38. **Sur.ly. 2012.** Sur.ly. [En línea] 2012. [Citado el: 14 de noviembre de 2014.] <http://sur.ly/o/sistedit.com/>.
39. **The Apache Software Foundation. 1999.** Apache JMeter. [En línea] 1999. [Citado el: 25 de abril de 2015.] <http://jakarta.apache.org/jmeter/>.
40. **VanDyk, John . 2011.** *An Introduction to Drupal Architecture*. DrupalCamp Des Moines. Iowa : s.n., 2011.
41. **Visual Paradigm for UML. 2013.** Visual Paradigm for UML. [En línea] 2013. [Citado el: 15 de noviembre de 2014.] <http://www.visual-paradigm.com/editions/enterprise.isp>.

## **Glosario de términos**

**Arquitectura:** Arquitectura del Software. Conjunto de elementos de programación adecuadamente estructurados dentro de un sistema, con el fin de crear una programación lógica y fiable para el diseño de aplicaciones.

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

**Script:** Es un guión o conjunto de instrucciones.

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

**SQL:** *Structured Query Language*: Lenguaje Estructurado de Petición. Es un lenguaje de programación, especializado para realizar consultas a bases de datos. Hoy en día, se puede considerar un estándar para la realización de consultas en multitud de SGBD.

**GPL:** La Licencia Pública General GPL por sus siglas en inglés, es una licencia creada por la *Free Software Foundation*, que está orientada principalmente a proteger la libre distribución, modificación y uso de software. Su propósito es declarar que el software cubierto por esta licencia es software libre y protegerlo de intentos de apropiación que restrinjan esas libertades a los usuarios.

**APIs:** Una interfaz de programación de aplicaciones o API (del inglés *Application Programming Interface*) es el conjunto de funciones y procedimientos (o métodos, en la programación orientada a objetos) que ofrece ciertas bibliotecas para ser utilizado por otro software como una capa de abstracción. Son usados generalmente en las bibliotecas.