

g r a t i s



UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS



INSTITUTO SUPERIOR POLITÉCNICO "JOSÉ ANTONIO ECHEVERRÍA"
FACULTAD DE INGENIERÍA INDUSTRIAL
CENTRO DE ESTUDIOS DE INGENIERÍA Y SISTEMA
INGENIERÍA EN INFORMÁTICA

Portal de La Revista Patria Grande

Trabajo de Diploma para optar por el Título de Ingeniería en Informática

Autor: Julio Cesar Isaza Vázquez

Tutor: Lic. Manuel Vázquez Acosta



Ciudad de La Habana

Junio del 2005

DECLARACIÓN DE AUTORÍA

Yo: Julio Cesar Isaza Vázquez me declaro como único autor de este trabajo y autorizo la Universidad de las Ciencias Informáticas (UCI) y al Centro de Estudios de Ingeniería y Sistemas (CEIS) para que hagan el uso de este trabajo de la manera que estimen conveniente.

Y para que así conste firmo la presente a los ____ días del mes de _____ del 2005.

Firma del Autor

Firma del Tutor

OPINIÓN DEL USUARIO DEL TRABAJO DE DIPLOMA

El Trabajo de Diploma, titulado: “**Portal de La Revista Patria Grande**”, fue realizado en la Universidad de Ciencias Informáticas. Este centro considera que, en correspondencia con los objetivos trazados, el trabajo realizado le satisface:

- Totalmente
- Parcialmente en un ____ %

Los resultados de este Trabajo de Diploma le reportan a esta Universidad los beneficios siguientes:

Como resultado de la implantación de este trabajo se reportará un efecto económico que asciende a <valor en MN o USD del efecto económico>

Y para que así conste, se firma la presente a los ____ días del mes de _____ del año _____

Representante de la entidad

Cargo

Firma

Cuño

OPINIÓN DEL TUTOR DEL TRABAJO DE DIPLOMA

Título: Portal de La Revista Patria Grande.

Autor: Julio Cesar Isaza Vázquez.

El tutor del presente Trabajo de Diploma considera que durante su ejecución el estudiante mostró las cualidades que a continuación se detallan.

<Aquí el tutor debe expresar cualitativamente su opinión y medir (usando la escala: muy alta, alta, adecuada) entre otras las cualidades siguientes:

- Independencia
 - Originalidad
 - Creatividad
 - Laboriosidad
 - Responsabilidad >

< Además, debe evaluar la calidad científico-técnica del trabajo realizado (resultados y documento) y expresar su opinión sobre el valor de los resultados obtenidos (aplicación y beneficios) >

Por todo lo anteriormente expresado considero que el estudiante está apto para ejercer como Ingeniero Informático; y propongo que se le otorgue al Trabajo de Diploma la calificación de <nota>. <Además, si considera que los resultados poseen valor para ser publicados, debe expresarlo también>

Firma

Fecha

Agradecimientos

...agradecer primeramente y antes que todo a mi familia, por apoyarme en estos 5 años de carrera, a mí mama Silvia Vazquez Escalona, mi abuela Esperanza Vega mis tías Inés y Gisela, mi tío Eduardo, Argelio, Hugo, mi padrastro Noel, mis primos, mi hermana en fin a todos, que por razones de espacio no incluiré el nombre; sin ellos no hubiese sido posible llegar hasta aquí.

...a todos mis amigos de la carrera por ser tan buenos compañeros, a los que se quedaron en Camagüey y a los que vinieron para la CUJAE en especial a Jose Fidalgo Hidalgo, Yudiel Alfred Tamayo Agramonte, Yanier Castellanos Gonzáles , Ronny Betancourt González, Manuel Alejandro Gil y Yanesky Montero Martínez.

...a mis amigos que no estudiaron conmigo pero que me ayudaron de una forma u otra, muchas gracias a todos ustedes.

...a mi tutor Lic. Manuel Vázquez Acosta por ayudarme realizar un trabajo mejor.

...a Lissette Soto Pelegrín por la atención y preocupación por nosotros aquí en la UCI.

... a todos los profesores que me impartieron clases a lo largo de la carrera por hacer de mi un mejor estudiante.

... a las instituciones universitarias que contribuyeron a mi formación como ingeniero, el Instituto Superior Politécnico José Antonio Echevarria, la Universidad de Ciencias Informáticas y en especial a la Universidad de Camagüey.

... a la buena música de Avril, Bon Jovi, Metallica, Buena Fé, Linkin' Park, Guns & Roses, Paulo FG, Bob Marley, etc, que me acompaño a lo largo de la carrera.

"Nunca consideres el estudio como una obligación sino como una oportunidad para penetrar en el bello y maravillosos mundo del saber."

Albert Einstein.

Dedicatoria

Este trabajo está dedicado a mi mamá Silvia Vázquez mis tías, Inés Vázquez, Gisela Vázquez, mi tío Eduardo Vázquez y mi abuela Esperanza Vega, gracias por apoyarme y siempre estar ahí cuando lo he necesitado, muchas gracias a todos, los quiero mucho....



Resumen

Difícilmente oímos hablar hoy en día de la era de la Televisión, la era del Teléfono, la era del Libro, y la era de la Internet. Son expresiones que pueden utilizarse en un breve momento en el tiempo, el momento inicial de la novedad de un invento, pero luego nos damos cuenta de que dicho invento sirve a una causa mayor: la Información y la Comunicación. Hoy por hoy, las estadísticas nos dicen que la búsqueda de información es el primer motivo, o la razón a la que alude el mayor número de navegantes, para utilizar la Internet. Por esta razón debemos gestionar y hacer el mejor uso posible de esa información en esta conocida como La Era de la Información. [18]

El Portal de La Revista Patria Grande de Venezuela es uno de los sitios, que brinda al mundo información relacionada con el proceso revolucionario y las transformaciones sociales llevadas a cabo en la República Bolivariana de Venezuela. Para dar solución a tan importante tarea se decidió el desarrollo de una aplicación Web capaz de actualizarse lo más fácil y rápidamente posible a través de un navegador, dándole la posibilidad al(los) editor(es) de la revista mantener actualizado el sitio.

Para desarrollar este sistema con las características deseadas se decidió hacer uso de un Sistema de Gestión Contenidos, que es un programa específicamente diseñado para administrar el contenido de un sitio Web, brindando la funcionalidad de servidor de portales.

Como resultado se obtuvo una aplicación Web muy fácil de actualizar y sin necesidad de que los editores dominen aspectos técnicos del desarrollo en ambientes Web el sistema, basada en roles, donde a cada persona que interactúa realiza una función específica, de acuerdo a los permisos que le haya asignado por el administrador del sistema.

Índice

INTRODUCCIÓN	1
Capítulo I. FUNDAMENTACIÓN DEL TEMA	6
1.1 INTRODUCCIÓN.....	6
1.2 APLICACIONES WEB	6
1.2.1 La World Wide Web (WWW). Un poco de Historia.....	6
1.2.2 Páginas estáticas Vs. dinámicas.....	7
1.2.3 Modelo Cliente-Servidor.....	8
1.2.4 Aplicaciones Web.....	9
1.3 SISTEMAS DE GESTIÓN DE CONTENIDOS.....	10
1.3.1 Historia de los CMS.....	11
1.3.2 ¿Qué entendemos por un CMS?	11
1.3.3 Categorías de la funcionalidad de los CMS.....	12
1.3.3.1 Creación de contenido.....	12
1.3.3.2 Gestión de contenido.....	13
1.3.3.3 Publicación.....	13
1.3.3.4 Presentación	13
1.4 OBJETO DE ESTUDIO	14
1.4.1 Descripción del proceso de negocio actual.....	15
1.4.2 Situación problemática.....	15
1.4.3 Problema.....	15
1.5 PROPUESTA DE SOLUCIÓN.....	15
1.6 FUNDAMENTACIÓN DE LOS OBJETIVOS DEL TRABAJO	15
1.6.1 Objetivo General.....	16
1.6.2 Objetivos Específicos	16
1.7 CONCLUSIONES.....	17
Capítulo II. TENDENCIAS Y TECNOLOGÍAS ACTUALES UTILIZADAS	18
2.1 INTRODUCCIÓN	18
2.2 SISTEMAS DE GESTIÓN DE CONTENIDOS.....	18
2.2.1 Necesidad de un CMS	18
2.2.2 Principales Ventajas.....	19
2.2.3 CMS comerciales y de código abierto	19
2.2.4 Criterios de Evaluación.....	20
2.2.5 Sistemas de Gestión de Contenidos en el mercado actual.....	21
2.3 FUNDAMENTACIÓN DE LAS TECNOLOGÍAS EN QUE SE BASA LA PROPUESTA	21
2.3.1 PHP.....	21
2.3.2 Lenguaje de consulta estructurado (SQL).....	22
2.4 FUNDAMENTACIÓN DE LA METODOLOGÍA A UTILIZAR.....	23
2.4.1 El proceso unificado del software (RUP).....	23
2.4.2 UML (Unified Modeling Language).....	24
2.4.1.1 Por que es necesario UML.....	25

2.5 HERRAMIENTAS QUE SE UTILIZARON.....	25
2.5.1 Rational Rose.....	25
2.5.2 Gestor de bases de Datos MySQL.....	26
2.5.2 Zend Studio.....	27
2.5.3 Xoops.....	27
2.5.3.1 Diseño modular, orientado a objetos.....	28
2.5.3.2 Un proceso de desarrollo organizado.....	28
2.5.4 Otras Herramientas utilizadas.....	29
2.6 PROPUESTA.....	29
2.7 CONCLUSIONES.....	30
Capítulo III. DESCRIPCIÓN DE LA PROPUESTA DE SOLUCIÓN.....	31
3.1 INTRODUCCION.....	31
3.2 DESCRIPCION DE LOS PROCESOS DEL NEGOCIO.....	32
3.3 MODELO DE DOMINIO.....	33
3.4 CAPTURA DE LOS REQUERIMIENTOS.....	36
3.4.1 Requerimientos funcionales del Sistema.....	36
3.4.2 Requerimientos no funcionales del Sistema.....	38
3.5 DESCRIPCION DEL SISTEMA PROPUESTO.....	40
3.6 ACTORES DEL SISTEMA.....	41
3.7 MODELO DE CASOS DE USO DEL SISTEMA.....	41
3.7.1 Diagrama de Caso de Usos.....	43
3.8 EXPANSIÓN DE LOS CASOS DE USO DEL SISTEMA.....	44
3.9 CONCLUSIONES.....	58
Capítulo IV. CONSTRUCCIÓN DE LA PROPUESTA DE SOLUCIÓN.....	59
4.1 INTODUCCIÓN.....	59
4.2 DIAGRAMA DE CLASES WEB.....	60
4.2.1 Paquete 1 Perfil de QuipusNews.....	63
4.2.2 Paquete 2 Perfil del Editor.....	64
4.2.2.1 Sub-Paquete 2.1 Manejar Contenido de Revista.....	64
4.2.2.2 Sub-Paquete 2.2 Manejar Contenido de Artículos.....	65
4.2.2.3 Sub-Paquete 2.3 Manejar Contenido de Noticias.....	65
4.2.3 Paquete 3 Perfil del Usuario Anónimo.....	66
4.2.3.1 Sub-Paquete 3.1 Mostrar Contenido de la Revista.....	66
4.2.3.2 Sub-Paquete 3.2 Mostrar Noticias.....	67
4.3 DISEÑO DE LA BASE DE DATOS.....	67
4.3.1 Clases persistentes.....	68
4.3.2 Diagrama del modelo de datos.....	69
4.4 PRINCIPIOS DE DISEÑO, MANEJO DE ERRORES Y ESTANDARES DE CODIFICACIÓN.....	70
4.4.1 Principios de Diseño.....	70
4.4.2 Manejo de Errores.....	71
4.6 MODELO DE DESPLIEGUE.....	75
4.7 MODELO DE IMPLEMENTACIÓN.....	75

4.8 CONCLUSIONES.....	84
Capítulo V. ESTUDIO DE FACTIBILIDAD.....	85
5.1 INTRODUCCIÓN.....	85
5.2 PLANIFICACIÓN.....	86
5.3 COSTOS.....	88
5.4 BENEFICIOS TANGIBLES E INTANGIBLES.....	89
5.5 ANÁLISIS DE COSTOS Y BENEFICIOS.....	90
CONCLUSIONES.....	91
RECOMENDACIONES.....	93
REFERENCIAS BIBLIOGRÁFICAS.....	94
BIBLIOGRAFÍA.....	96
GLOSARIO DE TÉRMINOS.....	99
ANEXOS.....	101
Anexo 1.....	101
Anexo 2.....	102
Anexo 3.....	103
Anexo 4.....	104
Anexo 5.....	106
Anexo 6.....	107
Anexo 7.....	108
Anexo 8.....	108

INTRODUCCIÓN

“Para quienes tenemos el privilegio de haber nacido en la América Latina o el Caribe, es un hecho indiscutible que pertenecemos a algo más que una circunstancia geográfica. Habitamos como lo advirtieron en su tiempo Bolívar y Martí, la Patria Grande destinada a despertar y unirse frente a los Estados Unidos ---que como definió el Libertador, <<parecen estar predeterminados por la providencia para plagar de hambre y miseria a la humanidad, en nombre de la libertad>>---, o detener el paso al gigante de las siete leguas, una de las más preclaras definiciones realizadas del imperialismo por el Héroe Nacional Cubano [1].

Corresponde a Patria Grande incorporar a sus páginas, con exactitud y alma, el acontecer diario de las misiones sociales que han unido, en toda la tierra bolivariana, el esfuerzo de venezolanos y cubanos, para garantizar la asistencia integral de salud a los ciudadanos que viven en los más remotos y pobres rincones del país, erradicar el analfabetismo y posibilitar la Revolución educacional, que hoy alcanza desde la enseñanza primaria a la Educación Superior.”[1].

La revista Patria Grande debido a su adolescencia carece de un portal digital para que los editores publiquen las ediciones mensuales y sean vistas en todo el mundo a través de la Internet por tanto surge una situación problemática debido a que unos de los objetivos fundamentales de la Revista es que el mundo conozca a través de Patria Grande la realidad Venezolana. El problema está dado en que no existe un portal digital capaz de gestionar, y organizar el contenido de la Revista.

La administración de contenido puede concebirse como una herramienta para poner ideas en práctica de la manera más directa posible. Añadir nuevo contenido a los sitios estáticos, especialmente en los ambientes de negocio donde el material se debe autorizar por varios departamentos, puede convertirse en una labor titánica. En cambio, con un Sitio Web administrado mediante un Sistema de Gestión de Contenidos, el nuevo contenido puede ser agregado fácilmente según lo requerido, en el mismo día si

es necesario, sin ningún costo, pudiendo además dar seguimiento a los resultados. El portal de la revista Patria Grande pretende hacer uso de un Sistema de Gestión de Contenidos, para publicar sus artículos y noticias, estos sistemas actualmente son muy utilizados en el mundo del desarrollo Web, pues resulta muy sencillo poner al día el contenido dinámico y diseño del portal de forma regular, sin la necesidad de adquirir las habilidades de un especialista.

Los gestores de contenidos proporcionan un entorno que posibilita la actualización, mantenimiento y ampliación de la Web con la colaboración de múltiples usuarios. En cualquier entorno virtual ésta es una característica importante, que además puede ayudar a crear una comunidad cohesionada que participe más de forma conjunta.

El sistema consiste en una interfaz que controla una o varias bases de datos donde se aloja el contenido del sitio, imágenes, textos, ficheros de audio, ficheros de video, etc. El sistema permite manejar de manera independiente el contenido por una parte y el diseño por otra. Así, es posible manejar el contenido y darle en cualquier momento un diseño distinto al sitio sin tener que darle formato al contenido de nuevo, además de permitir la fácil y controlada publicación en el sitio a varios editores.

Un aspecto crucial en el uso de esta tecnología, es que el contenido que se ingresa a un Sistema de Gestión de Contenidos es reutilizable. Almacenar información en una base de datos significa que su contenido puede ser reutilizado en muchos lugares dentro de su sitio Web y ajustado a diversos formatos para cualquier dispositivo: teléfono móvil, PDA e incluso para impresión. [2]

En la actualidad, aparte de la ampliación de las funcionalidades de los CMS, uno de los campos más interesantes es la incorporación de estándares que mejoran la compatibilidad de componentes, facilitan el aprendizaje al cambiar de sistema y aportan calidad y estabilidad. Algunos de estos estándares son CSS, que permite la creación de hojas de estilo; XML, un lenguaje de marcas que permite estructurar un documento;

XHTML, que es un subconjunto del anterior orientado a la presentación de documentos vía Web; WAI, que asegura la accesibilidad del sistema; y RSS, para syndicar contenidos de tipo noticia.

El Portal de la revista Patria Grande utiliza el Sistema de Gestión de Contenidos como una plataforma de publicación, aprovechando la estructura modular de estos sistemas, que permite personalizar el desenvolvimiento de nuevos módulos en este caso específico para atender las necesidades de la Revista además de contribuir a extender la plataforma.

Otro aspecto a tener en cuenta es la plataforma sobre la cual se desarrollara la aplicación, muchos países, entre ellos Venezuela y Cuba, han optado por el desarrollo de software libre, para evitar la dependencia tecnológica, y por la seguridad que brindan los software distribuidos con el código fuente (open source).

Con el nuevo Portal se espera como **aporte práctico** proveer al personal encargado de redactar los artículos para la revista, de una aplicación Web con las funcionalidades necesarias para publicar los mismos, además de brindarle una interfase de administración para la completa actualización y organización del contenido que se maneja en el mismo, sin necesidad de ningún conocimiento técnico por parte de los administradores u editores. Todo esto basándose en un sistema de usuarios y roles definidos en el sitio.

Por tanto el **objeto de estudio** de este trabajo se relaciona principalmente con la gestión de información en Internet, sus tendencias actuales, así como las soluciones más factibles a este problema.

El **campo de acción** se centra en los Sistemas de Gestión de Contenidos como herramientas para la gestión de información, así como sus tendencias actuales. Además se deben estudiar las plataformas de desarrollo Web más utilizadas, aplicaciones para Internet y software libres. La **Hipótesis** parte de la idea de que si se desarrolla una aplicación Web, basada en un Sistema de Gestión de Contenidos, es

posible lograr que el Portal de la Revista Patria Grande se mantenga actualizado con las ediciones mensuales de la revista.

Como **objetivo principal** se define la creación de una aplicación Web para dar solución al problema de la publicación del Portal de la Revista Patria Grande la cual lanza un publicación mensual, permitiendo al editor del portal publicar el contenido de forma rápida y sencilla sin la necesidad de conocimientos técnicos sobre HTML u otras técnicas de desarrollo Web.

Multiplicar el mensaje de la revista y hacerlo accesible a internautas en cualquier parte del mundo. Los materiales publicados en Patria Grande, de forma general, tienen un destinatario universal, de manera que se podrá importar todo el contenido del sumario a la página Web.

Utilizar las posibilidades de espacio, inmediatez y alcance de la Web para convertir el sitio en un medio de análisis e información sobre los principales acontecimientos que ocurren alrededor de las tres temáticas básicas que aborda *Patria Grande*: la integración cubano-venezolana, el ALBA y el proceso bolivariano.

El Portal tiene como **objetivos específicos** mostrar al usuario que visite el sitio Web una versión en formato digital de cada edición mensual de la revista Patria Grande; así mismo brindar versiones en formato PDF de cada artículo publicado en la revista así como de las revista en general.

Para cumplir estos objetivos se han llevado a cabo las siguientes tareas:

- Estudio y asimilación del entorno de trabajo.
- Identificar las necesidades de los clientes.
- Declarar los requisitos funcionales y no funcionales que debe cumplir el sistema.
- Análisis de cómo se encuentran en la arena internacional las tecnologías que se utilizan para llevar a cabo sistemas como el que se pretende desarrollar.
- Selección de la metodología de Análisis y Diseño de sistemas informáticos, que facilite la creación y garantice la calidad del sistema.
- Selección de las herramientas para llevar a cabo el proyecto y la elección de la plataforma en la que se desarrollará la aplicación. Fundamentando su elección.

- Desarrollar el Análisis y Diseño de la aplicación.
- Implementación de una aplicación Web que brinde la solución al problema planteado.

Se pretende finalmente obtener un producto de software a la altura de las exigencias actuales de producción de software en nuestra sociedad, acorde con los estándares internacionales de catalogación y los estándares de diseño y presentación de aplicaciones Web.

La estructura de este documento consta de cinco capítulos:

El primer capítulo aborda los fundamentos teóricos necesarios para la comprensión plena de los temas tratados en el resto del documento.

El segundo capítulo aborda la situación de las tecnologías a utilizar en el desarrollo de la aplicación, se comparan y seleccionan las mejores propuestas para el trabajo, y se explican los conceptos principales que se van a tratar.

El tercer capítulo está dedicado al análisis del modelo del negocio del Portal.

El cuarto capítulo abarca todo lo referente al diseño del sistema, aquí se definirá la estructura de la aplicación para dar respuesta al modelo del negocio.

El quinto y último capítulo está destinado al estudio de factibilidad.

Posteriormente se adicionan las páginas de conclusiones, bibliografía y glosario de términos.

Capítulo I.

FUNDAMENTACIÓN DEL TEMA

1.1 INTRODUCCIÓN

En este capítulo se brinda una vista global y detallada de los temas relacionados con las aplicaciones Web que manejan contenido dinámico, las características y ventajas del uso de los Sistemas de Gestión de Contenidos; las características de cada herramienta utilizada, así como la descripción de los principales conceptos asociados al dominio del problema que son necesarios para entender el modelo de negocio y la propuesta de solución.

En este capítulo se identifican los principales problemas que fundamentan la propuesta de solución, se marcan los objetivos generales y específicos, se describen los procesos del negocio relacionados con el objeto de estudio de este trabajo. También se hace un análisis sobre el comportamiento de las tecnologías punteras del mercado en esta área.

1.2 APLICACIONES WEB

1.2.1 La World Wide Web (WWW). Un poco de Historia.

La World Wide Web fue concebida en 1989 por un informático de la Organización Europea de Investigación Nuclear (CERN) llamado Tim Berners-Lee. Era un sistema de hipertexto para compartir información basado en Internet, concebido originalmente para servir como herramienta de comunicación entre los científicos nucleares del CERN. [3]

Hoy, la Web es algo cotidiano para una gran parte de los millones de usuarios de Internet que hay en todo el mundo. Sus utilidades son diversas, su impacto en la economía mundial es apreciable. No sólo hay documentos de texto: hay imágenes, vídeos, música, se pueden comprar cosas, se pueden hacer reservaciones etc. [3]

1.2.2 Páginas estáticas Vs. dinámicas

En la Web podemos encontrar, o construir, dos tipos de páginas:

1. Las que se presentan sin muchas funcionalidades más allá de los enlaces.
2. Las páginas que tienen animaciones u otros efectos gráficos y en las que podemos interactuar.

Las primeras páginas son las que denominamos páginas estáticas, se construyen con el lenguaje HTML. Estas páginas son muy sencillas de crear, aunque ofrecen pocas ventajas tanto a los desarrolladores como a los visitantes, ya que sólo se pueden presentar textos planos acompañados de imágenes y a lo sumo contenidos multimedia como pueden ser videos o sonidos.

El segundo tipo de páginas se denomina página dinámica. Una página es dinámica cuando se incluye cualquier efecto grafico o funcionalidad y para ello es necesario utilizar otros lenguajes de programación, aparte del simple HTML.

A pesar de que las páginas dinámicas nos puedan en un principio limitar a causa de su mayor complejidad con respecto a las estáticas, todas las ventajas que nos ofrecen compensan con creces este esfuerzo inicial. No obstante, hay que ser consciente del posible interés que pueda tener para uno el lanzarse en esta aventura de aprender un nuevo lenguaje y volver a rediseñar su propio sitio. [20]

Las aplicaciones dinámicas pueden ser escritas en varios lenguajes de programación y haciendo uso de plataformas de desarrollo existentes que proveen cada una al programador mejores o peores funcionalidades. [20]

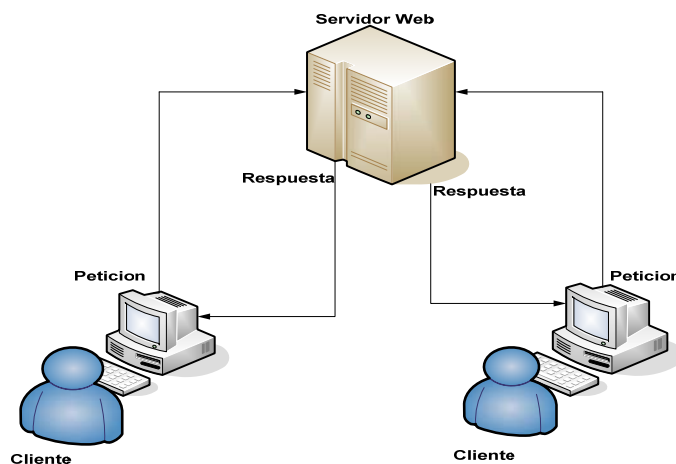
Debido a las evidentes ventajas de las páginas dinámicas, la construcción de aplicaciones Web se ha convertido en la tendencia principal de los desarrolladores de esta rama, gracias a la posibilidad que brindan de separar el contenido que se maneja de la forma de presentación del mismo y a su vez la facilidad del mantenimiento de dichas aplicaciones.

1.2.3 Modelo Cliente-Servidor.

El paradigma cliente-servidor es uno de los más extendidos dentro de los servicios a través de red. [19]

La arquitectura cliente-servidor llamado modelo cliente-servidor o servidor-cliente es una forma de dividir y especializar programas y equipos de cómputo a fin de que la tarea que cada uno de ellos realiza se efectúe con la mayor eficiencia, y permita simplificar las actualizaciones y mantenimiento del sistema. [19]

La idea básica y general que hay detrás de este modelo es que hay alguien que ofrece algo (el servidor) y alguien que quiere algo (el cliente). En el caso de las páginas Web tenemos un servidor Web que es aquél que tiene las páginas Web (o sea, la información) y un cliente (un navegador) que es el que pide la página Web (generalmente para mostrarla).



Modelo Cliente Servidor para la Web.

En esta arquitectura la capacidad de proceso está repartida entre el servidor y los clientes. En la funcionalidad de un programa distribuido se pueden distinguir 3 capas o niveles:

1. Manejador de Base de Datos (Nivel de almacenamiento),
2. Procesador de aplicaciones o reglas del negocio (Nivel lógico) y
3. Interfaz del usuario (Nivel de presentación)

De acuerdo con la distribución de la lógica de la aplicación hay dos posibilidades:

- Cliente delgado: si el cliente solo se hace cargo de la presentación.
- Cliente pesado: si el cliente asume también la lógica del negocio.

En la actualidad se suele hablar de arquitectura de tres niveles, donde la capa de almacenamiento y la de aplicación se ubican en uno o más servidores, conocidos como servidores de datos y servidores de aplicaciones. [19]

1.2.4 Aplicaciones Web.

El desarrollo de páginas Web se divide básicamente en dos vertientes, Diseño gráfico, Programación Web. Y actualmente, entró como nueva vertiente la Ingeniería del Software aplicada a la Web.

El diseño grafico se centra en la estética de la Web utilizando herramientas como Dreamweaver, Flash, Photoshop, herramientas WYSIWYG, etc.

La programación Web por su parte se centra en el desarrollo de aplicaciones Web, mediante diferentes tecnologías, los lenguajes etiquetados, las páginas activas y las base de datos.

Una aplicación Web no es más que una aplicación cliente-servidor que se apoya en la WWW, sus tecnologías, lenguajes y protocolos. Con la llegada de la Web, las

aplicaciones cliente/servidor han evolucionado hacia las aplicaciones Web donde toda la aplicación reside en un servidor Web.

Aplicación		Aplicación Web
Aplicación Cliente	Software específicamente programado, que el cliente tiene que tener instalado	El navegador de Internet. El cliente no tiene que instalar nada más, porque las respuestas del servidor' le llegan en forma de páginas Web.
Interfaz	Parecido a cualquier aplicación del sistema (p.ej: ventanas, menús, interfaz de comandos)	Páginas web programadas en HTML.
Aplicación Servidor	Software específicamente programado que debe estar instalado en el servidor.	Software específicamente programado que debe ser ejecutado por un servidor Web.
Protocolo.	Cualquier protocolo de comunicaciones (generalmente, basado en TCP/IP).	Protocolo http

Características generales de las aplicaciones cliente/servidor y las aplicaciones Web.

Actualmente, las aplicaciones están limitadas por tecnologías que no fueron diseñadas pensando en estas aplicaciones: HTML, HTTP, etc. El futuro: romper con estas limitaciones. XHTML, XML, SOAP, objetos distribuidos, etc.

1.3 SISTEMAS DE GESTIÓN DE CONTENIDOS

Realizar un Web puede ser un trabajo complicado y muy laborioso si no se dispone de las herramientas adecuadas. En los últimos años se ha desarrollado el concepto de sistema de gestión de contenidos (Content Management Systems o CMS). Se trata de herramientas que permiten crear y mantener un Web con facilidad, encargándose de los

trabajos más tediosos que hasta ahora ocupaban el tiempo de los administradores de las Web. [4]

1.3.1 Historia de los CMS

A principios de los años noventa, el concepto de sistemas de gestión de contenidos era desconocido. Algunas de sus funciones se realizaban con aplicaciones independientes: editores de texto y de imágenes, bases de datos y programación a medida.[4]

Ya en año 1994 Illustra Information Technology utilizaba una base de datos de objetos como repositorio de los contenidos de una Web, con el objetivo de poder reutilizar los objetos y ofrecía a los autores un entorno para la creación basado en patrones. La idea no cuajó entre el público y la parte de la empresa enfocada a la Web fue comprada por AOL, mientras que Informix adquirió la parte de bases de datos. [4]

RedDot es una de las empresas pioneras que empezó el desarrollo de un gestor de contenidos el año 1994. No fue hasta a finales del año siguiente que presentaron su CMS basado en una base de datos [4].

1.3.2 ¿Qué entendemos por un CMS?

CMS significa sistema de administración de contenido (en inglés Content Management System), un sistema de este tipo funciona para la creación y administración de contenido, actualmente se usa para denominar a los sistemas que sirven para administrar el contenido de páginas Web. [5]

El sistema consiste en una interfaz que controla una o varias bases de datos donde se aloja el contenido del sitio. El sistema permite manejar de manera independiente el contenido por una parte y el diseño por otra. Así, es posible manejar el contenido y darle en cualquier momento un diseño distinto al sitio sin tener que darle formato al contenido de nuevo, además de permitir la fácil y controlada publicación en el sitio a varios editores. Un ejemplo clásico es el de editores que cargan el contenido al

sistema y otro de nivel superior que permite que estos contenidos sean visibles a todo público. [5]

Un CMS tiene dos funciones principales: facilitar la creación de contenidos y la presentación de esos contenidos. Con respecto a la primera, provee una serie de herramientas para que publicar contenido sea tan fácil como rellenar un formulario, y haya, además, una sola fuente para todos ellos; con respecto a la segunda, facilita la publicación de contenidos en múltiples formatos a partir de una sola fuente, y añade metadatos a los mismos, para facilitar la navegación en múltiples facetas (temporal, por categorías o por autor, son sólo tres ejemplos posibles). También habría que considerar otras dos fases: gestión de contenidos y mantenimiento de los mismos; aunque estas fases se pueden incluir en la anterior. [6]

1.3.3 Categorías de la funcionalidad de los CMS.

James Robertson [7] propone una división de la funcionalidad de los sistemas de gestión de contenidos en cuatro categorías: creación de contenido, gestión de contenido, publicación y presentación. [4]

1.3.3.1 Creación de contenido

Un CMS aporta herramientas para que los creadores sin conocimientos técnicos en páginas Web puedan concentrarse en el contenido. Lo más habitual es proporcionar un editor de texto WYSIWYG, en el que el usuario ve el resultado final mientras escribe, al estilo de los editores comerciales, pero con un rango de formatos de texto limitado. Esta limitación tiene sentido, ya que el objetivo es que el creador pueda poner énfasis en algunos puntos, pero sin modificar mucho el estilo general del sitio Web. [4]

Hay otras herramientas como la edición de los documentos en XML, utilización de aplicaciones ofimáticas con las que se integra el CMS, importación de documentos existentes y editores que permiten añadir marcas, habitualmente HTML, para indicar el formato y estructura de un documento. [4]

Un CMS puede incorporar o no una o varias de estas herramientas, además podría proporcionar también un editor WYSIWYG por sus ventajas a la hora de editar contenido y la comodidad de acceso desde cualquier ordenador con un navegador y acceso a Internet. [4]

1.3.3.2 Gestión de contenido

Los documentos creados se depositan en una base de datos donde también se guardan el resto de datos de la Web, cómo son los datos relativos a los documentos (versiones hechas, autor, fecha de publicación y caducidad, etc.), datos y preferencias de los usuarios, la estructura de la Web, etc. [4]

La estructura de la Web se puede configurar con una herramienta que, habitualmente, presenta una visión jerárquica del sitio y permite modificaciones. Mediante esta estructura se puede asignar un grupo a cada área, con responsables, editores, autores y usuarios con diferentes permisos. Eso es imprescindible para facilitar el ciclo de trabajo (workflow) con un circuito de edición que va desde el autor hasta el responsable final de la publicación. El CMS permite la comunicación entre los miembros del grupo y hace un seguimiento del estado de cada paso del ciclo de trabajo. [4]

1.3.3.3 Publicación

Una página aprobada se publica automáticamente cuando llega la fecha de publicación, y cuando caduca se archiva para futuras referencias. En su publicación se aplica el patrón definido para toda la Web o para la sección concreta donde está situada, de forma que el resultado final es un sitio Web con un aspecto consistente en todas sus páginas. Esta separación entre contenido y forma permite que se pueda modificar el aspecto visual de un sitio Web sin afectar a los documentos ya creados y libera a los autores de preocuparse por el diseño final de sus páginas. [4]

1.3.3.4 Presentación

Un CMS puede gestionar automáticamente la accesibilidad del Web, con soporte de normas internacionales de accesibilidad como WAI, y adaptarse a las preferencias o

necesidades de cada usuario. También puede proporcionar compatibilidad con los diferentes navegadores disponibles en todas las plataformas (Windows, Linux, Mac, Palm, etc.) y su capacidad de internacionalización lo permite adaptarse al idioma, sistema de medidas y cultura del visitante. [4]

El sistema se encarga de gestionar muchos otros aspectos como son los menús de navegación o la jerarquía de la página actual dentro del Web, añadiendo enlaces de forma automática. También gestiona todos los módulos, internos o externos, que incorpore al sistema. Así por ejemplo, con un módulo de noticias se presentarían las novedades aparecidas en otro Web, con un módulo de publicidad se mostraría un anuncio o mensaje animado, y con un módulo de foro se podría mostrar, en la página principal, el título de los últimos mensajes recibidos. Todo eso con los enlaces correspondientes y, evidentemente, siguiendo el patrón que los diseñadores hayan creado. [4]

1.4 OBJETO DE ESTUDIO

El objeto de estudio de este trabajo se centra en el desarrollo de aplicaciones y tecnologías de desarrollo Web, y la gestión de contenido.

Con el objetivo de combatir las mentiras que algunos medios de comunicación intentan transmitir a los pueblos de América y al mundo acerca del proceso revolucionario bolivariano, y para difundir los logros del Convenio entre Cuba y Venezuela, la Alternativa Bolivariana para las Américas y el proceso revolucionario bolivariano, sale a la luz la revista Patria Grande que, como su nombre lo indica, se refiere a la unidad de toda Nuestra América, de la cual este convenio es precursor.

Para la lograr una gestión eficiente de la información que se desea manejar, se estudiarán los Sistemas de Gestión de Contenido, ya que constituyen una herramienta poderosa utilizada en la actualidad para este tipo de tarea.

1.4.1 Descripción del proceso de negocio actual

Actualmente la revista Patria Grande no cuenta con un portal digital capaz de publicar sus artículos en la Internet. Desde el lanzamiento del primer ejemplar de la revista en febrero del presente año solo cuenta con la versión editada en papel.

1.4.2 Situación problemática.

La revista Patria Grande, debido a su juventud, carece de un portal digital para que los editores publiquen las ediciones mensuales y sean vistas en todo el mundo a través de Internet. Esto trae como consecuencia que la revista solo este disponible a aquellos que tengan acceso a la edición impresa, restringiendo el alcance de dicha revista.

1.4.3 Problema

El problema está dado en que no existe un portal digital capaz de gestionar, y organizar el contenido de la revista.

1.5 PROPUESTA DE SOLUCIÓN

Tomando en cuenta que la revista Patria Grande carece de un portal digital para que los editores publiquen las ediciones mensuales se propone el desarrollo de una aplicación Web capaz de implementar las funcionalidades necesarias para mantener el portal actualizado.

Se decidió utilizar para gestionar la información un sistema manejador de contenidos debido a su flexibilidad, eficiencia, y fácil aprendizaje para los administradores del sitio, pues no es necesario que tengan conocimientos técnicos, de HTML u otro lenguaje o técnica de desarrollo Web.

1.6 FUNDAMENTACIÓN DE LOS OBJETIVOS DEL TRABAJO

Para llevar a cabo la solución propuesta se proponen un conjunto de objetivos.

1.6.1 Objetivo General

Como objetivo principal se define la creación de una aplicación Web para dar solución al problema de la publicación del Portal de la Revista Patria Grande la cual lanza una publicación mensual, permitiendo al administrador publicar el contenido de forma rápida y sencilla sin la necesidad de conocimientos técnicos sobre HTML u otras técnicas de desarrollo Web.

Multiplicar el mensaje de la revista y hacerlo accesible a internautas en cualquier parte del mundo. Los materiales publicados en Patria Grande, de forma general, tienen un destinatario universal, de manera que se podrá importar todo el contenido del Sumario a la página Web.

Utilizar las posibilidades de espacio, inmediatez y alcance de la Web para convertir el sitio en un medio de análisis e información sobre los principales acontecimientos que ocurren alrededor de las tres temáticas básicas que aborda *Patria Grande*: la integración cubano-venezolana, el ALBA y el proceso bolivariano.

1.6.2 Objetivos Específicos

Los objetivos específicos del portal son:

Permitir brindar al visitante una versión digital de de la revista Patria Grande del mes en curso así como la posibilidad de buscar ediciones de meses anteriores.

- Ofrecer a los internautas una versión digital de la revista Patria Grande.
- Ofrecer en formato PDF una versión de cada artículo publicado en la revista.
- Publicar noticias de diversos temas, principalmente relacionados con la integración Cubano-Venezolana.

1.7 CONCLUSIONES

En este capítulo se realiza una caracterización de las tecnologías actuales, así como un análisis de las mejores prácticas en la actualidad que se acomodan al objeto de estudio y la solución propuesta. Se definieron además los objetivos generales y específicos, según un análisis previo de la situación problemática y el problema originado en cuestión.

Capítulo II.

TENDENCIAS Y TECNOLOGÍAS ACTUALES UTILIZADAS

2.1 INTRODUCCIÓN

En el presente capítulo, se hace un análisis del estado actual de las tecnologías que pudieran ser adecuadas para la construcción del sistema que se pretende desarrollar. En algunos casos son necesarias comparaciones que fundamentarán la propuesta final.

2.2 SISTEMAS DE GESTIÓN DE CONTENIDOS

2.2.1 Necesidad de un CMS

En el capítulo anterior se han presentado bastantes motivos para ver la utilidad de un sistema que gestione un entorno Web, pero se podría pensar que no es necesario para un Web relativamente pequeña o cuando no se necesitan tantas funcionalidades. Eso sólo podría ser cierto para un Web con unas pocas páginas estáticas para el que no se prevea un crecimiento futuro ni muchas actualizaciones, lo que no es muy realista. En cualquier otro caso, la flexibilidad y escalabilidad que permiten estos sistemas, justifican su utilización en prácticamente cualquier tipo de Web.

Muchos usuarios particulares utilizan CMS gratuitos para elaborar y gestionar sus Webs personales, obteniendo Webs dinámicos llenos de funcionalidades. El resultado que obtienen es superior al de algunas empresas que se limitan a tener páginas estáticas que no aportan ningún valor añadido.

2.2.2 Principales Ventajas

Accesibilidad: posibilidades de administrar o contenido utilizando o Internet Explorer desde cualquier computador conectado a Internet en red local, sin necesidad de instalar software adicional. [8]

Interfase: Totalmente basado en el concepto de "Skins", proporciona gran agilidad para la actualización del sitio o portal, tanto de características visuales (diseño) como de su contenido. [8]

Facilidades de Menú de Usuario: con una interfaz intuitiva, o panel de control muy fácil de usar para que cualquier persona, sin grandes conocimientos de Internet, pueda administrar el contenido. Todo el contenido es manejado por el administrador por medio de un poderoso editor HTML basado en el concepto WYSIWYG (What You See Is What You Get). [8]

Interactividad con los Visitantes: proporciona una clasificación de usuarios, configuración para evitar envío de malas palabras, foros, envío de postales, Chat y mensajeros instantáneos, propiciando interactividad del sitio con los usuarios y visitantes. [8]

Flexibilidad: Por poseer una estructura modular, permite personalizar el desenvolvimiento de nuevos módulos específicamente para atender las necesidades de su empresa, tornando su sitio en una poderosa herramienta de trabajo e interacción con sus visitantes y clientes. [8]

2.2.3 CMS comerciales y de código abierto

Se puede hacer una primera división de los CMS según el tipo de licencia. Por una parte están los CMS comercializados por empresas que consideran el código fuente un activo más que tienen que mantener en propiedad, y que no permiten que terceros tengan acceso. Por la otra tenemos los de código fuente abierto, desarrollados

por individuos, grupos o empresas que permiten el acceso libre y la modificación del código fuente. [4]

La disponibilidad del código fuente posibilita que se hagan personalizaciones del producto, correcciones de errores y desarrollo de nuevas funciones. Este hecho es una garantía de que el producto podrá evolucionar incluso después de la desaparición del grupo o empresa creadora.

Algunas empresas también dan acceso al código, pero sólo con la adquisición de una licencia especial o después de su desaparición. Generalmente las modificaciones sólo pueden hacerlas los mismos desarrolladores, y siempre según sus prioridades. [4]

Los CMS de código abierto son mucho más flexibles en este sentido, pero se podría considerar que la herramienta comercial será más estable y coherente al estar desarrollada por un mismo grupo. En la práctica esta ventaja no es tan grande, ya que los CMS de código abierto también están coordinados por un único grupo o por empresas, de forma similar a los comerciales. [4]

2.2.4 Criterios de Evaluación.

Previamente la organización debe conocer cuáles son sus necesidades contestando a las típicas preguntas:

- ¿Qué tipo de contenido quiero gestionar?
- ¿Documentos ofimáticos, páginas Web, sitios Web, sitios de comercio electrónico, informes, gráficos, audio, vídeo?
- ¿Qué quiero hacer con el contenido?
- ¿Publicarlo en portales corporativos, en mi intranet?
- ¿El contenido va a variar con frecuencia?
- ¿Son facturas, por ejemplo, que no van a cambiar nunca?
- ¿Necesito controlar lo que publico o sólo gestionar su publicación?
- ¿Quién creará el contenido?
- ¿Cualquier usuario puede aportarlo?

Una vez despejadas estas dudas procedemos a evaluar algunos aspectos generales y particularidades que, podrían, son esenciales para posicionar adecuadamente un sistema de Gestión de Contenidos. Tabla 1 Criterios de Evaluación.

2.2.5 Sistemas de Gestión de Contenidos en el mercado actual.

La gama de soluciones CMS que existen en el mercado es muy amplia. Actualmente se cuenta con gran variedad de sistemas manejadores de contenidos, desarrollados en diferentes lenguajes de programación, donde los más populares son los desarrollados en las plataformas Java 2 Enterprise Edition (J2EE) y los desarrollados en PHP.

Algunos como el Xoops, Mambo, Drupal, Typo3, PHPNuke, Xaraya, PostNuke, phpWebSite, y otros más, entran en el grupo de PHP, otros como el EXOPlatform, el Apache Lenya, InfoGlue, y el LifeRay pertenecen al grupo de J2EE.

También podemos encontrar algunos desarrollados en otros lenguajes tales como Phyton, Parser-3, Perl y C#, tal es el caso del Rainbow desarrollado bajo la plataforma .Net totalmente gratis y de código abierto, pero que no tiene tanta popularidad como los antes mencionados.

2.3 FUNDAMENTACIÓN DE LAS TECNOLOGÍAS EN QUE SE BASA LA PROPUESTA

2.3.1 PHP

PHP, (acrónimo de "PHP: Hypertext Preprocessor"), es un lenguaje interpretado de alto nivel embebido en páginas HTML. La mayoría de su sintaxis es similar a C, Java y Perl, con solamente un par de características específicas. [9]

Al nivel más básico, PHP puede hacer cualquier cosa que se pueda hacer con un script CGI, como procesar la información de formularios, generar páginas con contenidos dinámicos, o mandar y recibir cookies. [9]

Una de sus características más potentes es su soporte para gran cantidad de bases de datos. Entre su soporte pueden mencionarse InterBase, MSSQL, MySQL, Oracle, Informix, PostgreSQL, ODBC, Unix dbm, FrontBase entre otras más. [9]

PHP es la opción natural para los programadores en máquinas con Linux que ejecutan servidores Web con Apache, pero funciona igualmente bien en cualquier otra plataforma de UNIX o de Windows, con el software de Netscape o del Web server de Microsoft. PHP también utiliza las sesiones de HTTP, conectividad de Java, expresiones regulares, LDAP, SNMP, IMAP, POP3, protocolos de COM (bajo Windows). [9]

Debido a su amplia distribución PHP está perfectamente soportado por una gran comunidad de desarrolladores. Como producto de código abierto, PHP goza de la ayuda de un gran grupo de programadores, permitiendo que los fallos de funcionamiento se encuentren y se reparan rápidamente. El código se pone al día continuamente con mejoras y extensiones de lenguaje para ampliar las capacidades de PHP. [9]

2.3.2 Lenguaje de consulta estructurado (SQL).

Debido a la diversidad de lenguajes y de bases de datos existentes, la manera de comunicar entre unos y otros sería realmente complicada de gestionar de no ser por la existencia de estándares que nos permiten el realizar las operaciones básicas de una forma universal. [10]

El lenguaje de consulta estructurado (SQL) es un lenguaje de base de datos normalizado, utilizado por los diferentes motores de bases de datos para realizar determinadas operaciones sobre los datos o sobre la estructura de los mismos. Pero como sucede con cualquier sistema de normalización hay excepciones para casi todo; de hecho, cada motor de bases de datos tiene sus peculiaridades y lo hace diferente de otro motor, por lo tanto, el lenguaje SQL normalizado (ANSI) no nos servirá para resolver todos los problemas, aunque si se puede asegurar que cualquier sentencia escrita en ANSI será interpretable por cualquier motor de datos. [10]

Actualmente, está en marcha un proceso de revisión del lenguaje por parte de los comités ANSI e ISO, que debería terminar en la definición de lo que en este momento se conoce como SQL3. Las características principales de esta nueva encarnación de SQL deberían ser su transformación en un lenguaje stand-alone (mientras ahora se usa como lenguaje hospedado en otros lenguajes) y la introducción de nuevos tipos de datos más complejos que permitan, por ejemplo, el tratamiento de datos multimediales. [10]

2.4 FUNDAMENTACIÓN DE LA METODOLOGÍA A UTILIZAR

La tendencia actual en el software lleva a la construcción de sistemas más complejos y eficientes. Queremos un que software que este mejor adaptado a nuestras necesidades pero esto a su vez, simplemente hace el software más complejo. También lo queremos más rápido.

Conseguirlo sin embargo es difícil, el problema del software se reduce a la dificultad que afrontan los desarrolladores para coordinar las múltiples cadenas de trabajo de un gran proyecto de software. La comunidad de desarrolladores estas necesitada de una forma coordinada de trabajar. Necesitan un proceso que integre las múltiples facetas del desarrollo. Necesitan un método común.

2.4.1 El proceso unificado del software (RUP).

El Proceso Unificado de Desarrollo es el resultado de más de 30 años de experiencia, es una solución al problema del software. [11]

En primer lugar, el Proceso Unificado es un proceso de desarrollo de software. Un proceso de desarrollo de software es el conjunto de actividades necesarias para transformar los requisitos de un usuario en un sistema de software. Sin embargo el Proceso Unificado es más que un simple proceso; es un marco de trabajo genérico que puede especializarse para una gran variedad de sistemas software, para diferentes áreas de aplicación, diferentes tipos de organizaciones, diferentes niveles de aptitud y diferentes tamaños de proyecto. [11]

El Proceso Unificado utiliza el Lenguaje Unificado de Modelado (Unified Modeling Language, UML) para preparar todos los esquemas de un sistema software. De hecho UML es un aparte esencial del Proceso Unificado, sus desarrollos fueron paralelos. [11]

No obstante, los verdaderos aspectos definitorios del Proceso Unificado se resumen en tres frases claves, dirigido por casos de uso, centrado en la arquitectura, e iterativo e incremental. Esto es lo que hace único al Proceso Unificado. [11]

2.4.2 UML (Unified Modeling Language).

UML (Unified Modeling Language) es un lenguaje que permite modelar, construir y documentar los elementos que forman un sistema software orientado a objetos. Se ha convertido en el estándar de facto de la industria, debido a que ha sido impulsado por los autores de los tres métodos más usados de orientación a objetos: Grady Booch, Ivar Jacobson y Jim Rumbaugh. En el proceso de creación de UML han participado, no obstante, otras empresas de gran peso en la industria como Microsoft, Hewlett-Packard, Oracle o IBM, así como grupos de analistas y desarrolladores. [12]

2.4.1.1 Por que es necesario UML.

Uno de los objetivos principales de la creación de UML era posibilitar el intercambio de modelos entre las distintas herramientas CASE orientadas a objetos del mercado. Para ello era necesario definir una notación y semántica común. [13]

Conforme aumenta la complejidad del mundo, los sistemas informáticos también deberán crecer en complejidad. En ellos se encuentran diversas piezas de hardware y software que se comunican a grandes distancias mediante una red, misma que está vinculada a bases de datos que, a su vez, contienen enormes cantidades de información. Si desea crear sistemas que lo involucren con este nuevo milenio ¿como manejará tanta complejidad? La clave está en organizar el proceso de diseño de tal forma que los analistas, clientes, desarrolladores y otras personas involucradas en el desarrollo del sistema lo comprendan y convengan con él. El UML proporciona tal organización. [13]

La necesidad de diseños sólidos ha traído consigo la creación de una notación de diseño que los analistas, desarrolladores y clientes acepten como pauta (tal como la notación en los diagramas esquemáticos sirve como pauta para los trabajadores especializados en electrónica). El UML es esa misma notación. [13]

2.5 HERRAMIENTAS QUE SE UTILIZARON

2.5.1 Rational Rose

Es la herramienta líder en el mundo para el modelado de sistemas complejos y de tiempo real. Para los analistas de negocios, Rose ofrece la capacidad de modelar y visualizar sus procesos de negocios y destacar oportunidades para aumentar la eficiencia. [14]

Para los analistas de datos, el modelado de su diseño de base de datos en Rose, mejora la comunicación entre usted y los desarrolladores, y, cuando usted modela sus

casos de uso en Rose, asegura que la solución sea creada con el usuario en mente. [14]

Rational Rose unifica a los analistas de negocios, sistemas y datos al permitirles crear y administrar modelos en una herramienta con un solo lenguaje de modelado. [14]

2.5.2 Gestor de bases de Datos MySQL.

MySQL es una de las bases de datos más populares desarrolladas bajo la filosofía de código abierto. La desarrolla y mantiene la empresa MySQL AB pero puede utilizarse gratuitamente y su código fuente está disponible. [15]

Inicialmente, MySQL carecía de elementos considerados esenciales en las bases de datos relacionales, tales como integridad referencial y transacciones. A pesar de ello, atrajo a los desarrolladores de páginas Web con contenido dinámico, justamente por su simplicidad; aquellos elementos faltantes fueron llenados por la vía de las aplicaciones que la utilizan. [15]

Poco a poco los elementos faltantes en MySQL están siendo incorporados tanto por desarrollos internos, como por desarrolladores de software libre. Entre las características disponibles en las últimas versiones se puede destacar:

- Amplio subconjunto del lenguaje SQL. Algunas extensiones son incluidas igualmente.
- Disponibilidad en gran cantidad de plataformas y sistemas.
- Diferentes opciones de almacenamiento según si se desea velocidad en las operaciones o el mayor número de operaciones disponibles.
- Transacciones y claves foráneas.
- Conectividad segura.
- Replicación.
- Búsqueda e indexación de campos de texto.

Según las cifras del fabricante, existirían cinco millones de copias de MySQL corriendo en la actualidad, lo que supera la base instalada de cualquier otra herramienta de bases de datos. [15]

2.5.2 Zend Studio

Se trata de un programa de la casa Zend, orientado a desarrollar aplicaciones Web, en lenguaje PHP. El programa, además de servir de editor de texto para páginas PHP, proporciona una serie de ayudas que pasan desde la creación y gestión de proyectos hasta la depuración de código. [16]

Zend Studio consta de dos partes en las que se dividen las funcionalidades de parte del cliente y las del servidor. Las dos partes se instalan por separado, la del cliente contiene el interfaz de edición y la ayuda. Permite además hacer depuraciones simples de scripts, aunque para disfrutar de toda la potencia de la herramienta de depuración habrá que disponer de la parte del servidor, que instala Apache y el módulo PHP o, en caso de que estén instalados, los configura para trabajar juntos en depuración. [16]

2.5.3 Xoops

XOOPS es un portal dinámico orientado a objetos escrito en PHP y con MySQL como base de datos y soporte de almacenamiento, haciendo de XOOPS una herramienta ideal para desarrollar sitios Web de pequeñas y grandes dimensiones, sitios internos de empresas, sitios corporativos, weblogs y mucho más. Si bien comenzó como un sistema de portal, XOOPS actualmente está en el camino de los sistemas de manejo de contenidos. El mismo puede servir como marco de trabajo en sitios pequeños, medianos y grandes.[17]

XOOPS significa eXtensible Object Oriented Portal System, es decir, Sistema extensible de portales orientado a objetos. Es una herramienta construida con herramientas de largo alcance, que puede escalarse para adaptarse a tus necesidades. XOOPS se ofrece bajo los términos de la Licencia Pública General (GPL) GNU, lo cual

significa que puedes usarlo y modificarlo gratuitamente. Puedes modificar y alterar el programa todo lo que necesites o quieras, en tanto te atengas a los términos de distribución de la GPL. [17]

2.5.3.1 Diseño modular, orientado a objetos

XOOPS básicamente consiste en, el programa núcleo (core) de XOOPS, que provee el marco operativo y administrativo básico, una serie de módulos oficiales que cubren las necesidades básicas de un sitio dinámico y tantos módulos opcionales como puedas necesitar para disponer de funcionalidades adicional. Es posible agregar nuevas funciones a través de un sencillo proceso de instalación de módulos. Esta aproximación modular hace que XOOPS sea altamente personalizable. Sólo tienes que instalar los módulos que consideres útiles, y puedes retirarlos en cualquier momento, no hay necesidad de cargar tu sitio con código innecesario, y puedes configurarlo para que tenga sólo aquello que en verdad necesitas. [17]

2.5.3.2 Un proceso de desarrollo organizado

XOOPS es, claramente, un producto 'vivo'. XOOPS está siendo activamente desarrollado y extendido a través de un proceso organizado que incluye la consulta pública con los miembros de la comunidad XOOPS. Se han establecido ocho equipos para asegurar que el sistema evolucione de manera coherente y consistente. Los equipos son:

- Desarrollo del núcleo
- Desarrollo de módulos
- Desarrollo de temas
- Control de calidad
- Soporte a la comunidad
- Documentación
- Relaciones comunitarias
- Desarrollo de producto

Los equipos de desarrollo y la comunidad XOOPS en general están continuamente mejorando el sistema y extendiendo su funcionalidad, con énfasis en la producción de código de alta calidad. [17]

2.5.4 Otras Herramientas utilizadas

- Macromedia Dreamweaver para el diseño de las páginas Web.
- Adobe Photoshop CS para el trabajo con imágenes.
- Adobe ImageReady CS para el trabajo con imágenes.

2.6 PROPUESTA

Finalmente se propone el desarrollo de un portal digital utilizando el sistema manejador de contenidos XOOPS por las ventajas y características antes expuestas de estos sistemas.

Como ambiente de desarrollo integrado (IDE, siglas en ingles) se propone el Zend Studio por ser un potente editor de código en lenguaje PHP

Como gestor de Bases de datos se propone el MySQL por sus características y por la completa integración de PHP con el mismo.

2.7 CONCLUSIONES

Este capítulo realizó un análisis de las tendencias y tecnologías actuales que se pretenden utilizar en el desarrollo del proyecto. Asimismo se ha valorado y justificado el uso de cada herramienta y tecnología.

Llegando a la conclusión de que el lenguaje de programación a utilizar será PHP, como gestor de base de datos, MySQL, y el Sistema Manejador de Contenidos XOOPS.

Capítulo III.

DESCRIPCIÓN DE LA PROPUESTA DE SOLUCIÓN.

3.1 INTRODUCCIÓN

En el presente capítulo se hace la descripción de la propuesta que trae este trabajo, para ello se describen los procesos del negocio que tiene que ver con el objeto de estudio, de acuerdo a esto se llega a la conclusión que debido a la poca estructuración de esos procesos, para poder entender el contexto en que se emplaza el sistema necesitamos definir conceptos que podemos agrupar en un Modelo de Dominio, para capturar correctamente los requisitos y poder construir un sistema correcto.

Además se enumeran los requisitos funcionales y no funcionales que debe tener el sistema que proponemos, lo que permite hacer una concepción general del sistema, e identificar mediante un Diagrama de Caso de Uso, las relaciones de los actores que interactúan con el sistema, y las secuencias de acciones con las que interactúan.

3.2 DESCRIPCIÓN DE LOS PROCESOS DEL NEGOCIO.

Para describir los procesos del negocio que se relacionan con el campo de acción de este trabajo, se hace necesario comprender el proceso editorial de la revista Patria Grande y sus objetivos.

El primer paso dentro del modelado del negocio es la identificación de los diferentes procesos del negocio de la organización que en este caso sería la casa editorial de Patria Grande.

Patria Grande es una revista impresa mensual, que aborda temáticas relacionadas con la integración cubano-venezolana, la Alternativa Bolivariana para las Américas (ALBA) y el proceso revolucionario que encabeza el Presidente Hugo Chávez.

La búsqueda de información.

Este trabajo propone una nueva alternativa en el proceso de la búsqueda de información acerca de la integración cubano-venezolana, la Alternativa Bolivariana para las Américas (ALBA) y el proceso revolucionario bolivariano.

El proyecto de llevar las páginas de Patria Grande a la Web tiene dos objetivos de índole editorial:

Multiplicar el mensaje de la revista y hacerlo accesible a los internautas en cualquier parte del mundo. Los materiales publicados en Patria Grande, de forma general, tienen un destinatario universal, de manera que se podrá importar todo el contenido del Sumario a la aplicación Web.

Utilizar las posibilidades de espacio, inmediatez y alcance de la Web para convertir el sitio en un medio de análisis e información sobre los principales acontecimientos que ocurren alrededor de las tres temáticas básicas que aborda Patria Grande: la integración cubano-venezolana, el ALBA y el proceso bolivariano.

Mantenimiento de la información.

Mantener el contenido del portal es una tarea realizada por el editor del portal, el mismo se encargara de configurar los artículos y las noticias.

La forma de adicionar contenido al portal dependerá de su tipo (artículo, noticias, imágenes).

Los artículos se adicionan mensualmente, correspondientes a la edición mensual de la revista, por otra parte las noticias e imágenes se actualizaran de forma diaria o esporádicamente.

Con la propuesta de este trabajo se trataría de brindar la mayor flexibilidad posible para la publicación del contenido en el portal, se trataría además de que todos los usuarios que tengan acceso al las publicaciones mensuales y a los demás servicios del portal.

3.3 MODELO DE DOMINIO.

Debido a la sencillez de la estructura y los mecanismos en el proceso de administración y actualización del portal, llegamos a la conclusión de que el negocio que se está estudiando no necesita un modelado completo del negocio. Además como la creación del portal es una nueva alternativa en el proceso de búsqueda de información acerca de la integración cubano-venezolana, no existen reglas de negocio definidas para la actualización del mismo.

Por las características antes expuestas se utilizara un modelo de dominio con el cual pretendemos contribuir a la comprensión del contexto del sistema, y por lo tanto también contribuir a la comprensión de los requerimientos del sistema que se desprenden de este contexto. Para capturar correctamente los requerimientos y poder construir correctamente un sistema se necesita tener un firme conocimiento del funcionamiento del objeto de estudio del mismo.

El objetivo del modelo de domino es comprender y describir las clases más importantes dentro del contexto del sistema, en otras palabras el modelado del dominio deberá contribuir a una comprensión del problema que el sistema resuelve en relación a su contexto.

También nos auxiliaremos de un glosario de términos sobre los nombres para que identificar todos los conceptos que se utilizarán en el diagrama. El glosario y el modelo del dominio nos ayudaran a los desarrolladores, usuarios, clientes y otros interesados a utilizar un vocabulario común.

Se entenderá por **Artículos** a los materiales editoriales creados por la casa editorial de Patria Grande en calidad de artículos.

Se entenderá por **Noticias** a los materiales editoriales creados por la casa editorial de Patria Grande en calidad de noticias.

Se entenderá por **Revista** al material mensual donde se publicaran artículos y noticias, está constituida por una portada y una presentación editorial y los artículos publicados en ella.

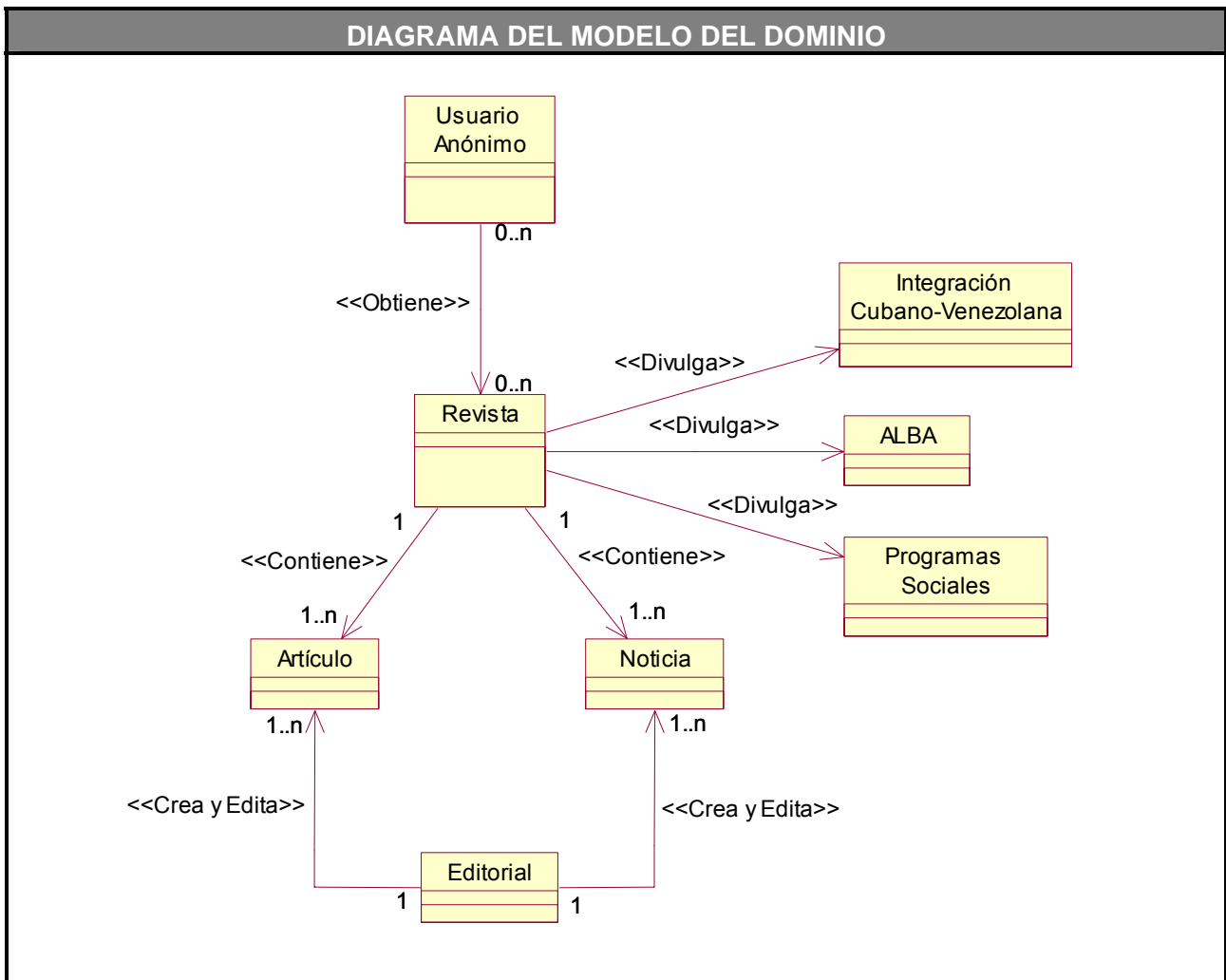
Se entenderá por **Editorial** a la casa editorial de Patria Grande.

Se entenderá por **Usuario Anónimo** a cualquier usuario que desee obtener la información publicada en la revista Patria Grande.

Se entenderá por **ALBA** a la Alternativa Bolivariana para las Américas, propuesta por el gobierno venezolano para hacer frente al tratado de libre comercio (ALCA) propuesto por los Estados Unidos de América.

Se entenderá por **Integración Cubano-Venezolana** a la mutua colaboración entra las dos hermanas naciones.

Se entenderá por **Programas Sociales** a los programas sociales llevados a cabo por la Republica Bolivariana de Venezuela.



3.4 CAPTURA DE LOS REQUERIMIENTOS.

“El propósito fundamental del flujo de trabajo de los requisitos es guiar el desarrollo hacia el sistema correcto. Esto se consigue mediante una descripción de los requisitos del sistema (es decir, las condiciones o capacidades que le sistema debe cumplir) suficientemente buena como para que pueda llegarse a un acuerdo entre el cliente (incluyendo los usuarios) y los desarrolladores sobre qué debe y que no debe hacer el sistema.”[11]

3.4.1 Requerimientos funcionales del Sistema.

De acuerdo con los objetivos planteados el sistema debe ser capaz de:

R1 Recibir contenido del sistema encargado de las ediciones.

- 1) El sistemas tiene que se capaz de recibir y adicionar los artículos enviados de QN.
- 2) El sistemas tiene que se capaz de recibir y adicionar las noticias enviados de QN.

R2 Administrar las revistas.

- 3) Crear Revista.
- 4) Editar Revista.
- 5) Eliminar Revista.

R3 Administrar artículos publicados en la revista.

- 6) Configurar artículos en la revista (asignarle la revista a la que pertenece).
- 7) Eliminar artículo de la revista.

R4 Administrar Secciones.

- 8) Crear una sección.
- 9) Editar una sección.
- 10) Eliminar una sección.

R5 Administrar contenido de las noticias publicadas en el portal.

- 11) Configurar noticia del portal (definirle la fecha de publicación y la fecha en que expira).
- 12) Eliminar noticia del portal.

R6 Administrar usuarios y roles del sistema.

- 13) El sistema debe ser capaz de permitir acceso al editor, administrador de la revista así como a los usuarios anónimos.

El editor puede:

- a) Crear y configurar la revista, es decir establecer la portada, presentación editorial, encuestas, imágenes etc.
- b) Configurar los artículos es decir ubicarlos en la revista a la que pertenecen.
- c) Configurar las noticias es decir establecer tiempo de vida (fecha de publicación y expiración).

El administrador puede:

- a) Establecer nuevos roles al sistema.
- b) Eliminar o adicionar usuarios al sistema.

El usuario anónimo puede:

- a) Acceder al contenido publicado en el portal.
- b) Descargar contenido del portal.

R7 Mostrar Contenido de la Revista.

- 14) Mostrar la portada de la revista.
- 15) Mostrar la presentación editorial de la revista.
- 16) Mostrar artículos de la revista.

R8 Mostrar Noticias.

- 17) Mostrar noticias que no hayan expirado.

R9 Mostrar ediciones anteriores de la revista.

- 18) El portal debe tener ubicado en lugar visible el acceso a las ediciones anteriores.
- 19) Debe mostrar todos sus artículos, con acceso desde el sumario.

R10 Mostrar colección de artículos Dossier.

- 20) El portal debe tener ubicado en un lugar visible el acceso a la colección de artículos dossier.
- 21)

R11 Mostrar sección de imágenes.

- 22) Mostrar imágenes de la galería.
- 23) Mostrar imágenes para coleccionar como fondo de pantalla.
- 24) Mostrar caricatura diaria.

R12 Mostrar enlaces con otros sitios de interés para la revista.

R13 Mostrar sección para las encuestas.

25) Debe mostrar la encuesta definida para el portal con acceso a los resultados de la misma y la posibilidad de votar en ella.

3.4.2 Requerimientos no funcionales del Sistema.

Los requerimientos no funcionales son propiedades o cualidades que el producto debe tener. Debe pensarse en estas propiedades como las características que hacen al producto atractivo, usable, rápido o confiable.

Apariencia o interfaz externa:

- Diseño orientado a llamar la atención del usuario y con una navegación sencilla y escalable.
- Construcción de enlaces rápidos o anclas para los documentos muy largos.

Usabilidad:

- El sistema podrá ser usado por cualquier persona que posea conocimientos básicos en el manejo de la computadora y de un ambiente Web en sentido general.

Rendimiento:

- Tiempos de respuestas rápidos al igual que la velocidad de procesamiento de la información.

Soporte:

- Se requiere un servidor de bases de datos con las siguientes características:
- Soporte para grandes volúmenes de datos y velocidad de procesamiento.
- Tiempo de respuesta rápido en accesos concurrentes.
- Versión de PHP 4.3.0 o superior.
- Servidor Web Apache 1.3.x o superior (se acepta IIS).
- Por parte del cliente se requiere un navegador capaz de interpretar JavaScript.

Portabilidad:

- Necesidad de que el sistema sea multiplataformas.

Seguridad:

- Identificar al usuario antes de que pueda realizar cualquier acción sobre el contenido del portal.
- Garantizar que la información sea publicada únicamente por quien tiene derecho a publicarla.
- Garantizar que las funcionalidades del sistema se muestren de acuerdo al nivel de usuario que este activo.
- Protección contra acciones no autorizadas o que puedan afectar la integridad de los datos.
- Verificación sobre acciones irreversibles (eliminaciones).

Político – Culturales:

- Debido a que se trata de un producto para la nación de Venezolana, el sistema debe ajustarse a las características político-culturales de este país.

Legales:

- El sistema se basa en un estándar de catalogación que se rige por normas internacionales.
- La plataforma escogida para el desarrollo de la aplicación, está basada en la licencia GNU/GPL.

Confiableidad:

- La herramienta de implementación a utilizar tiene soporte para recuperación ante fallos y errores.

Funcionalidad:

- Reducir al mínimo el tiempo en que carga el portal.
- Guardar en caché páginas de contenido para agilizar la navegación del portal.

Software:

- Navegador compatible o superior con Internet Explorer 4, o NetsCape Navigator.
- Macromedia Dreamweaver MX 2004.
- Adobe PhotoShop 7.0.
- MySQL 2.3.x.
- Apache 1.3.x.49
- PHP 4.3.4

3.5 DESCRIPCIÓN DEL SISTEMA PROPUESTO.

Para cumplir exitosamente con los objetivos propuestos en este trabajo y teniendo en cuenta los requerimientos planteados el sistema propone el desarrollo un módulo, el portal Patria Grande, para brindar la edición mensual de la revista así como las ediciones anteriores, además se incorpora el módulo de administración del sistema el cual no modelaremos ya que es incorporado por el propio Sistema Manejador de Contenido que se usara como plataforma para el portal.

Se considera la existencia de tres roles, el rol de administrador del sistema que se encargará de la configuración del Portal, el rol de editor general que se encargará de la creación de nuevas revistas y de la organización y publicación de los artículos y noticias así como actualización de la galería de imágenes y los ficheros en formato PDF asociados cada articulo de la revista y por ultimo el rol de usuario anónimo el cual podrá navegar por el portal obteniendo la información que necesite.

El módulo del portal podrá ser usado por cualquier usuario anónimo que desee obtener información de la versión digital de Patria Grande. Este podrá navegar por el portal como estime conveniente, participar en encuesta, descargar artículos en fin, beneficiarse de los servicios que ofrece el portal.

El modulo de administración controla todo lo que relacionado con la configuración del sistema, se usará para mantener el portal actualizado y en perfecto estado, solo tendrán acceso los administradores del sistema.

En resumen el sistema propuesto pretende crear una plataforma de publicación de contenido para las ediciones mensuales de Patria Grande así como la publicación para las principales noticias y comentarios relacionados con los temas de interés para Patria Grande.

3.6 ACTORES DEL SISTEMA.

ACTORES	JUSTIFICACIÓN
Usuario Anónimo	Representa a una persona que va a utilizar el sistema para buscar información sobre alguna temática determinada.
QuipusNews	Representa al sistema encargado de editar y enviar al portal los artículos y las noticias que serán publicados en cada revista.
Editor	Representa a una persona que tiene la capacidad de configurar la revista y realizar los cambios que estime convenientes.
Administrador	Representa a una persona que configura y controla el comportamiento del sistema, controlar a los usuarios, definir roles etc.

3.7 MODELO DE CASOS DE USO DEL SISTEMA.

Utilizando las facilidades que brinda el UML, se representarán los requisitos funcionales del sistema mediante un diagrama de casos de uso. Cada caso de uso especifica una secuencia de acciones que el sistema puede llevar a cabo interactuando con sus actores, incluyendo alternativas dentro de la secuencia. A continuación se presentan los casos de uso determinados para satisfacer los requerimientos funcionales del sistema:

CU-1	Recibir contenido del sistema encargado de las ediciones.
Actor	QuipusNews
Descripción	El sistema recibe el contenido enviado por QuipusNews.
Referencia	R1.

CU-2	Gestionar contenido del portal.
Actor	Editor
Descripción	El editor interactúa con el sistema para configurar el contenido del portal.
Referencia	R2, R3, R4, R5

CU-3	Administrar el sistema.
Actor	Administrador.
Descripción	El administrador accede al portal para configurar la seguridad.
Referencia	R6.

CU-4	Mostrar contenido de la revista.
Actor	Usuario Anónimo, Administrador, Editor.
Descripción	Un usuario solicita ver el contenido de la revista del mes.
Referencia	R7.

CU-5	Mostrar noticias publicadas en el portal.
Actor	Usuario Anónimo, Administrador, Editor.
Descripción	Un usuario solicita ver el contenido de las noticias publicadas en el portal.
Referencia	R8.

CU-6	Mostrar ediciones de revistas anteriores.
Actor	Usuario Anónimo, Administrador, Editor.
Descripción	Un usuario solicita ver una edición pasada de la revista.
Referencia	R9.

CU-8	Mostrar colección de artículos Dossier.
Actor	Usuario Anónimo, Administrador, Editor.
Descripción	Se muestra la colección de artículos dossier que hay en el portal.
Referencia	R10.

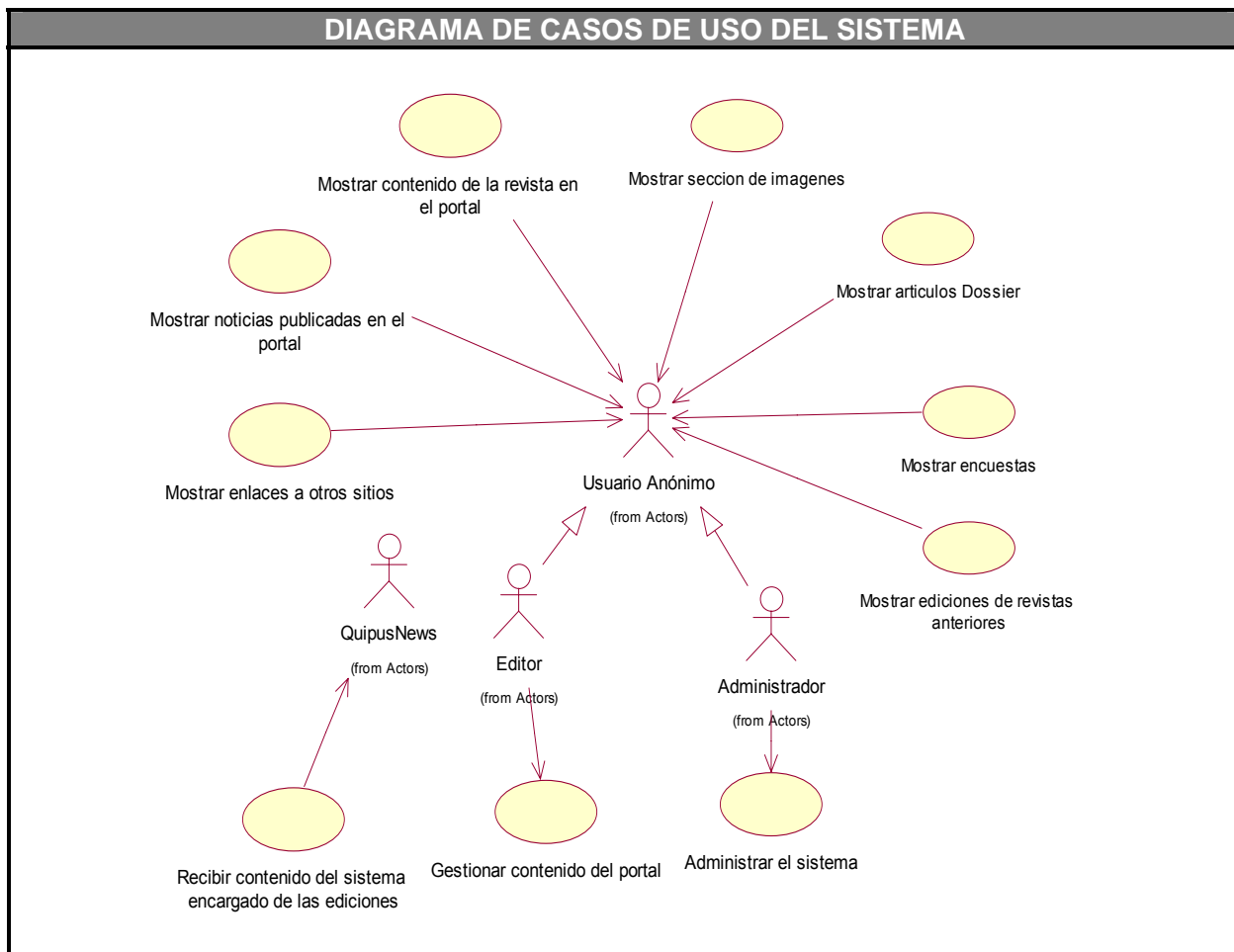
CU-9	Mostrar enlaces a otros sitios.
Actor	Usuario Anónimo, Administrador, Editor.
Descripción	Un usuario solicita ver los enlaces a otros sitios de interés para la revista los cuales soportan el contenido del portal.
Referencia	R11.

CU-10	Mostrar sección de imágenes.
Actor	Usuario Anónimo, Administrador, Editor.
Descripción	Muestra las secciones de imágenes existentes en el portal, cricaruta diaria galería de imágenes y cartel para coleccionar.
Referencia	R12.

CU-11	Mostrar sección para las encuestas.
Actor	Usuario Anónimo, Administrador, Editor.
Descripción	Muestra la encuesta publicada con posibilidad de participar de ella.
Referencia	R13.

3.7.1 Diagrama de Caso de Usos.

A continuación se representa el diagrama que muestra la relación existente entre los actores y los casos de uso:



3.8 EXPANSIÓN DE LOS CASOS DE USO DEL SISTEMA.

A continuación se representa la extensión de los casos de uso del sistema que se consideran de mayor importancia, el resto se adicionan en la sección de anexos (ver **Anexo 2, Anexo 3 y Anexo 4**):

CU-1	<i>Recibir contenido del sistema encargado de las ediciones</i>
Actores	QuipusNews.
Propósito	Que el sistema sea capaz de recibir el contenido enviado por el sistema QuipusNews por el protocolo HTTP.
Resumen	El caso de uso inicia cuando el sistema QuipusNews envía algún material al portal. El sistema QuipusNews envía materiales editados (artículos y noticias) listos para publicar a través del protocolo HTTP y el método POST el sistema debe ser capaz de recibir estos materiales, parsear los datos, e insertarlos en las tablas correspondientes de la base de datos que da soporte al sistema y en caso de los ficheros (imágenes y ficheros PDF) en los directorios correspondientes.
Referencias	R1
Precondiciones	
Poscondiciones	El contenido recibido debe quedar correctamente almacenado en la base de datos que da soporte al sistema, así como ficheros asociados en el envío.
Requisitos especiales	Para poder recibir cualquier material o fichero el envío debe autenticarse mediante un código encriptado el cual se chequea en el sistema. Luego de verificar la autenticación e insertar correctamente los datos el sistema debe enviar un mensaje a QuipusNews de reportando si la operación de recibir se realizó satisfactoriamente o si hubo algún problema.
Prototipo	

CU-2	<i>Gestionar contenido del portal</i>
Actores	Editor
Propósito	Que el Editor del portal sea capaz de mantener y configurar el contenido que será publicado en el portal.
Resumen	Este caso de uso inicia cuando el usuario accede al portal como Editor (<i>Pantalla 1</i>) para manipular y administrar el contenido de las revistas (<i>Pantalla 2</i>). Para el caso de las revistas se mostrara un listado con las ediciones de la revista existentes (<i>Pantalla 3-A</i>) donde el editor podrá crear una revista (<i>Pantalla 3-B</i>) correspondiente a una nueva edición, eliminar una revista existente (<i>Pantalla 3-C</i>) o mediante el enlace Editar (<i>Pantalla 3-D</i>) modificar los

datos de una revista existente (*Pantalla 4*). Para los artículos se presentara una lista con los existentes (*Pantalla 5*) donde podrá cambiarle el estado de publicación a publicado (*Pantalla 5-A*) y eliminar un artículo existente (*Pantalla 5-B*), además mediante el enlace Editar (*Pantalla 5-C*) posibilitara otras configuraciones como asignarle la revista a la que pertenece (*Pantalla 6-A*), asignarle la sección (*Pantalla 6-B*) si es que pertenece a alguna, y poner al artículo la categoría de dossier (*Pantalla 6-C*). Para el caso de las noticias se presenta un listado con las noticias existentes (*Pantalla 7*) donde el editor podrá eliminar una noticia existente (*Pantalla 7-A*) si lo desea, además mediante el enlace Editar (*Pantalla 7-B*) podrá configurarles la fecha a partir de la cual se van a publicar (*Pantalla 8-A*) y la fecha en la que van a expirar (*Pantalla 8-B*). Para las secciones se mostrara el listado de las secciones existentes (*Pantalla 9*) blindándole la posibilidad al editor de agregar una nueva sección (*Pantalla 9-A*), eliminar las secciones que desee (*Pantalla 9-B*), y mediante el enlace Editar (*Pantalla 9-C*) modificar los datos de las secciones (*Pantalla 10*).

Referencias

R2, R3, R4, R5

Precondiciones

Deben existir revistas y secciones creadas, también deben existir artículos y noticias en la base de datos.

Poscondiciones

El contenido debe quedar configurado o modificado según lo dispuesto por el editor y los cambios deben ser guardados en la base de datos.

Requisitos especiales

Prototipo



Ingresar

Nombre:

Contraseña:

¿No registrado? Presione **Aquí**.

Pantalla 1.



Pantalla 2

Revistas

Nro	Edicion	Editar	Borrar
1	Enero		X
2	Febrero		X
3	Marzo		X
4	Abril		X
5	Mayo		X
6	Junio		X

Adicionar Revista

Nro.

Edicion

Título del Dossier

Imagen Portada

Texto Portada

Imagen Prologo

Texto Prologo

Firma Prologo

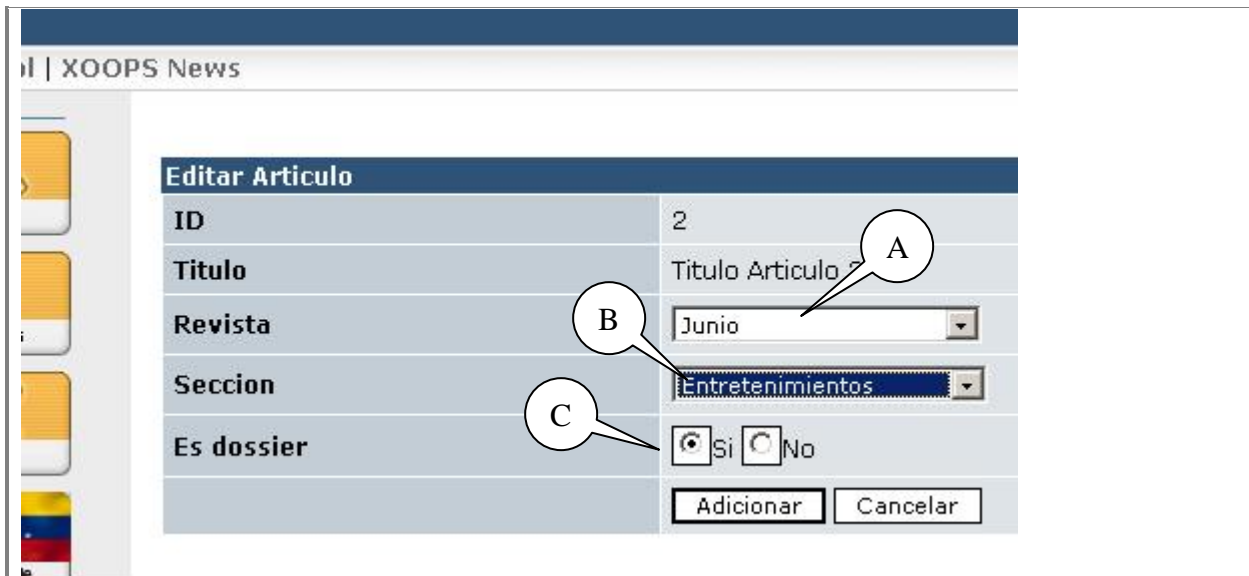
Secciones Cultura Tips Correo de Abrazos Entretenimientos Hoja Literaria

Pantalla 3

Pantalla 4

ID	Revista	Seccion	Titulo	Autor	Pub	Fecha_Edicion	Editar	Borrar
2	6	4	Titulo Articulo 2	ycg	✓	2000-12-12		X
1	6	4	Titulo Articulo 1	a	✓	2005-04-11		X
3	6	2	Titulo Articulo 3	a	✓	2004-12-12		X
4	6	3	Titulo Articulo 4	asd	✗	0000-00-00		X
5	6	4	Titulo Articulo 5	j	✗	0000-00-00		X
6	6	0	Discurso del Comandante en Jefe	El Administrador	✓	1983-11-14		X

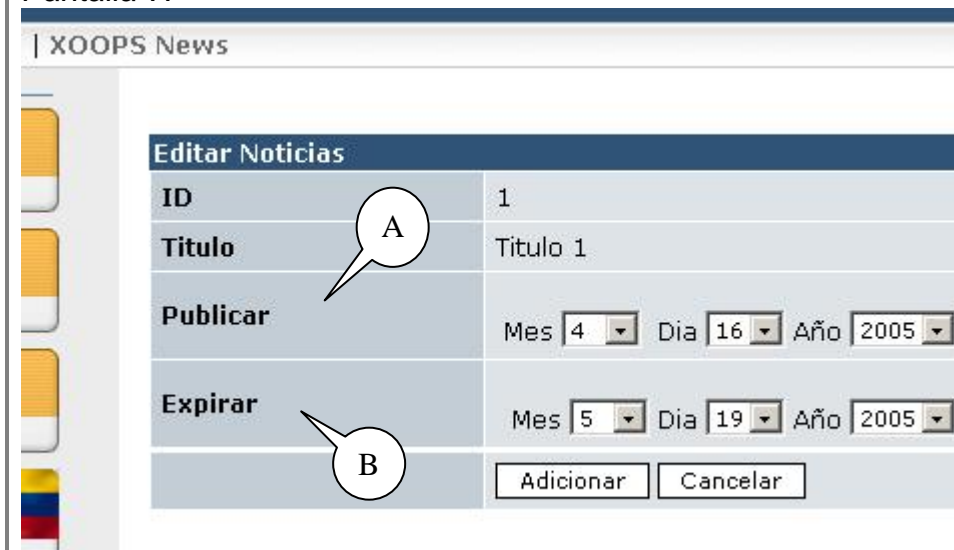
Pantalla 5



Pantalla 6.



Pantalla 7.



Pantalla 8.

Secciones				
Nro	Nombre	Descripción	Editar	Borrar
1	Cultura	seccion de deportes		X
2	Tips	tip de esta seccion		X
3	Correo de Abrazos	Descripción		X
4	Entretenimientos	Descripción		X
5	Hoja Literaria	Descripción		X

Adicionar Secciones

Nro.

Nombre

Descripción

Path: **body**

Adicionar

Pantalla 9

Editar

Nro.

Nombre

Descripción

Path: **body**

Adicionar

Pantalla 10

CU-3	<i>Administrar el sistema</i>
<p>Actores</p> <p>Propósito</p> <p>Resumen</p> <p>Este caso de uso inicia cuando un usuario accede al portal como administrador (<i>Pantalla 1</i>) permitiéndole ver los usuarios del sistema (<i>Pantalla 2-A</i>) crear nuevos usuarios (<i>Pantalla 2-B</i>), eliminar usuarios (<i>Pantalla 2-C</i>) o modificar los datos de un usuario (<i>Pantalla 2-D</i>), ver los grupos existentes (<i>Pantalla 3-A</i>), crear nuevos grupos de usuarios (<i>Pantalla 3-B</i>), modificar los grupos (<i>Pantalla 3-C</i>) y eliminar los grupos creados (<i>Pantalla 3-D</i>), permite además la administración general del portal (<i>Pantalla 4</i>).</p> <p>Referencias</p> <p>Precondiciones</p> <p>Poscondiciones</p> <p>Requisitos especiales</p>	<p>Administrador.</p> <p>Que le permita al administrador controlar la seguridad del portal, controlar los accesos, roles y usuarios.</p> <p>R6</p>
Prototipo	
 <p>Ingresar</p> <p>Nombre: administrador</p> <p>Contraseña: ●●●●●●●●</p> <p>Ingresar</p> <p>¿No registrado? Presione Aquí.</p> <p><i>Pantalla 1</i></p>	

Editar/Borrar Usuarios

Apodo	jisaza
	<div style="border: 1px solid black; padding: 2px;"> Modificar Usuario Modificar Usuario Borrar Usuario </div>

Add User

Apodo	<input type="text"/>
Nombre	<input type="text"/>
Email	<input type="text"/>
	<input type="checkbox"/> Permitir a otros Usuarios Ver esta dirección de Email
URL	<input type="text"/>
Zona de Tiempo	(GMT) Hora del Meridiano de Greenwich , Dublín, Edimburgo, Londres
ICQ	<input type="text"/>
AIM	<input type="text"/>
YIM	<input type="text"/>
MSNM	<input type="text"/>
País/Localidad	<input type="text"/>
Ocupación	<input type="text"/>
Intereses	<input type="text"/>
Firma	<div style="border: 1px solid black; height: 40px; width: 100%;"></div> <input type="checkbox"/>
	<input type="checkbox"/> Mostrar Firma
Forma de mostrar los Comentarios	Vista escalonada
Orden de los Comentarios	Viejos Primero
Método de Notificación: Cuando usted monitorea ej. un foro, como le gustaría recibir notificaciones de actualizaciones?	Mensaje Privado
Modo de Notificación por defecto	Notificarme de todas las actualizaciones seleccionadas
Información Adicional	<div style="border: 1px solid black; height: 40px; width: 100%;"></div> <input type="checkbox"/>
Rango	<input type="text"/>
Contraseña	<input type="text"/>
Reingrese la Contraseña	<input type="text"/>
¿ Recibir ocasionalmente noticias via email enviadas por los administradores y moderadores ?	<input type="radio"/> Si <input checked="" type="radio"/> No
	<input type="button" value="Enviar"/>

Pantalla 2

Editar Grupos

Editar Grupos	
Administradores	Modificar
Usuarios Registrados	Modificar
Usuarios Anónimos	Modificar
Nuevo Grupo	Modificar Borrar

Crear Nuevo Grupo

Nombre:

Descripción:

Privilegios System Admin

Avatares Banners Administrar Bloques Comentarios Buscar Usuarios Administrar Grupos
 Manejador de Imágenes Mail a Usuarios Administrar Módulos Preferencias Caritas
 Templates Rango de Usuarios Editar Usuarios

Privilegios Admin

Revistas Votaciones Álbum de fotos Enlaces

Privilegios Acceso

Álbum de fotos Humor Gráfico

Bloque de acceso

Izquierda

Contenidos en espera (ID: 4) Recomiéndenos (ID: 6) Top Envios (ID: 8) Comentarios Recientes (ID: 10) Opciones de Notificación (ID: 11) Portada (ID: 84) Foto al azar (ID: 117) Fotos recientes (ID: 118) Fotos Recientes con miniaturas (ID: 120) Últimas Fotos con miniaturas (ID: 121) Humor Gráfico (ID: 122) Enlaces Recientes (ID: 123) Top Enlaces (ID: 124) Menú del Usuario (ID: 1)

Centro

Prologo (ID: 89) Artículos (ID: 90)

Derecha

Ingreso (ID: 2) Noticias (ID: 112) Las mejores fotos (ID: 119) Buscar (ID: 3) Menú (ID: 5) ¿Quién está en línea? (ID: 7) Seleccionar Themes (ID: 12) Votaciones (ID: 115) Nuevos Miembros (ID: 9) Seleccion (ID: 97) Dossier (ID: 125)

Pantalla 3.

MENU

- SISTEMA
- VOTACIONES
- ENLACES

System Configuration

Avatares	Banners	Administrar Bloques	Comentarios	Buscar Usuarios
Administrar Grupos	Manejador de Imágenes	Mail a Usuarios	Administrar Módulos	Preferencias
Caritas	Templates	Rango de Usuarios	Editar Usuarios	

Pantalla 4.

<p>CU-4</p>	<p><i>Mostrar contenido de la revista en el portal</i></p>
<p>Actores</p>	<p>Usuario Anónimo, Administrador, Editor.</p>
<p>Propósito</p>	<p>Que el usuario sea capaz de ver el contenido publicado en el portal correspondiente a la revista del mes.</p>
<p>Resumen</p>	
<p>Este caso de uso inicia cuando un usuario accede al portal como usuario anónimo en la Internet o una intranet, entonces este se encarga de mostrar la revista correspondiente al mes en curso (<i>Pantalla 1</i>), la portada (<i>Pantalla 1-A</i>), la presentación editorial (<i>Pantalla 1-B</i>) y el sumario de los artículos asociados a ella (<i>Pantalla 1-C</i>).</p>	
<p>Referencias</p>	<p>R7</p>
<p>Precondiciones</p>	<p>Deben existir revistas creadas, así como artículos asociados además deben estar en estado de publicado, deben existir físicamente los ficheros asociados (imágenes y PDF) en sus directorios correspondientes.</p>
<p>Poscondiciones</p>	
<p>Requisitos especiales</p>	
<p>Prototipo</p>	
 <p>The screenshot shows the website for 'PATRIA LA REVISTA DE LA INTEGRACIÓN CUBANO-VENEZOLANA'. The page is for the April 2005 issue (No. 4). It features a red header with the magazine title and navigation links. The main content area is divided into sections: 'ESTE NÚMERO' with a portrait of Dausell Valdés and a short bio; a featured article '¿POR QUÉ NACEMOS?' with a detailed introduction; a 'sumario' (table of contents) listing various articles; and a 'Réplica' sidebar with a list of topics and a 'Galería de fotos' section.</p>	
<p><i>Pantalla 1</i></p>	

CU-5	<i>Mostrar noticias publicadas en el portal</i>
Actores	Usuario Anónimo, Administrador, Editor
Propósito	Que el usuario sea capaz de ver el contenido de las noticias publicadas en el portal.
Resumen	Este caso de uso inicia cuando un usuario accede al portal como usuario anónimo en la Internet o una intranet, entonces este se encarga de mostrar un sumario de las noticias (Pantalla 1-A) que estén publicadas en ese momento en el portal.
Referencias	R8
Precondiciones	
Poscondiciones	
Requisitos especiales	

Prototipo



Pantalla 1

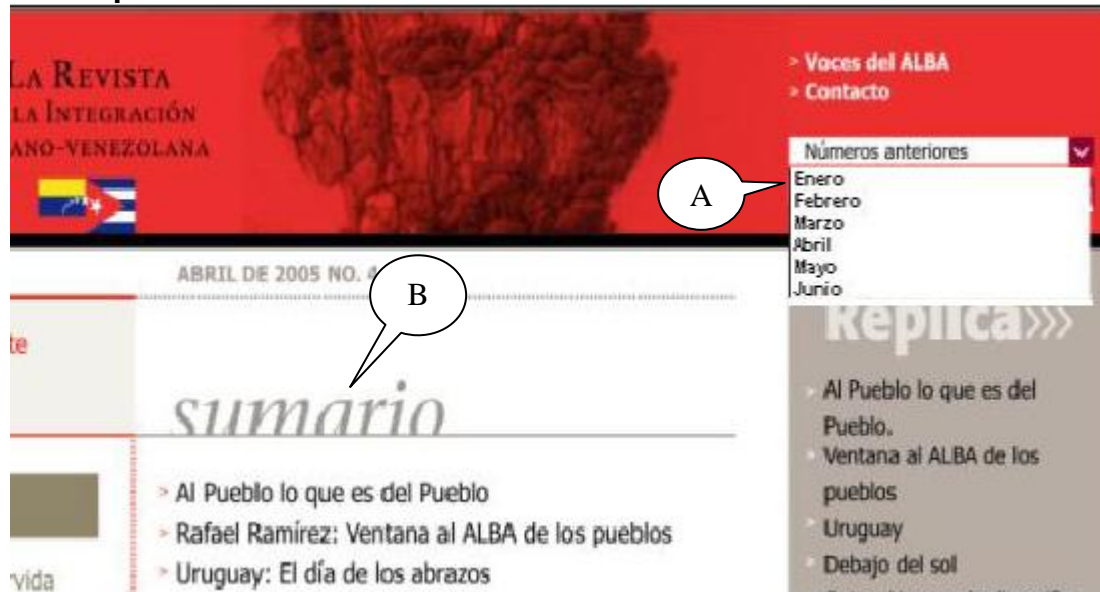
CU-6	<i>Mostrar ediciones de revistas anteriores.</i>
Actores	Usuario Anónimo, Administrador, Editor
Propósito	Que el usuario pueda seleccionar y ver alguna edición anterior de la revista.
Resumen	

Este caso de uso inicia cuando un usuario accede al portal como usuario anónimo en la Internet o una intranet, entonces se muestra al usuario las ediciones pasadas de la revista (Pantalla 1-A), entonces podrá acceder desde el sumario (Pantalla 1-B) a los artículos correspondientes a dicha edición.

Referencias
Precondiciones
Poscondiciones
Requisitos especiales

R9

Prototipo



Pantalla 1.

CU-7	<i>Mostrar artículos Dossier</i>
Actores	Usuario Anónimo, Administrador, Editor
Propósito	Que el usuario pueda ver y acceder a la colección de artículos dossier del portal.
Resumen	Este caso de uso inicia cuando un usuario accede al portal como usuario anónimo en la Internet o una intranet, entonces este se encarga de mostrar la colección de artículos dossier del portal (Pantalla 1) dando la posibilidad de acceder a su contenido (Pantalla 2).
Referencias Precondiciones Poscondiciones Requisitos especiales	R10
Prototipo	

LA REVISTA DE LA INTEGRACIÓN CUBANO-VENEZOLANA

ABRIL DE 2005 NO. 4

> Voces del ALBA
> Contacto

Números anteriores
Expedientes especiales
Dossier número 1
Dossier número 2
Dossier número 3
Dossier número 3

icidio o la fórmula del golpe de Estado se ni un cargamento más de Petróleo".

sumario

- > Al Pueblo lo que es del Pueblo
- > Rafael Ramírez: Ventana al ALBA de los pueblos
- > Uruguay: El día de los abrazos
- > India: Debajo del sol todo tiene su hora
- > Qatar: Venezuela diversifica su mercado petrolero

Los pueblos

ustrial-petrolero que lleva un país adentro. esidente Chávez al poder, los intereses de Venezuela (PDVSA) en un estado dentro

Pantalla 1.

PÁGINA INICIO | FEBRERO DE 2004 NO.1

Abril 2002, Contragolpe y Victoria Popular DOSSIER

¿Qué sucedió en Venezuela entre el 11 y 13 de abril de 2002?

La llegada del Presidente Chávez al poder, los intereses transnacionales convirtieron a Petróleos de Venezuela (PDVSA) en un estado dentro del Estado, gerenciada por una oligarquía que era cada vez más rica, mientras más desigual e injusta fuera la

Feliz López
Periodista corresponsal en Cuba.

DESCARGAR

Pantalla 2.

3.9 CONCLUSIONES.

Partiendo del análisis de los procesos del negocio, en este capítulo hemos definido los requerimientos que debe tener el sistema para un exitoso desarrollo, los cuales se representan mediante un Diagrama de Casos de Uso, así mismo se describieron las acciones que debe realizar cada actor que interactúa con el sistema.

Con los requerimientos y las funciones que han sido consideradas como necesarias para la construcción del sistema se puede dar paso al comienzo de la construcción del mismo.

Capítulo IV.

CONSTRUCCIÓN DE LA PROPUESTA DE SOLUCIÓN.

4.1 INTODUCCIÓN.

En este capítulo se modelan los artefactos necesarios para la construcción de aplicaciones Web. Los componentes de la aplicación son tratados como clases, y mediante la utilización de UML se podrán representar a través de diagramas de clases Web. Así mismo se presenta la propuesta del modelo de datos la cual nos posibilitara la construcción de la base de datos que servirá como soporte al sistema. Este capitulo además de modelar la lógica del negocio mediante las clases Web, y trata algunos principios del diseño del sistema.

4.2 DIAGRAMA DE CLASES WEB.

El diagrama de clases para las Aplicaciones Web, difiere un poco del resto de las aplicaciones que se construyen, puesto que en ellas son más importantes la modelación de la lógica y estado del negocio que los detalles de presentación. Dado que para el desarrollo de este sistema se utilizó el Sistema Manejador de Contenidos Xoops daremos una breve explicación del funcionamiento interno del mismo para una mejor comprensión de los diagramas de clases.

Como se dijo en el Capítulo 2 del documento, XOOPS básicamente consiste en, el programa núcleo (core) de XOOPS, que provee el marco operativo y administrativo básico, es posible agregarle nuevas funcionalidades instalando nuevos módulos que hayan sido desarrollados por terceros, el módulo de instalación añade automáticamente las entradas en la tabla de configuración basándose en el contenido del fichero `xoops_version.php` (obligatorio para cada módulo), el cual se utiliza para generar las opciones de configuración de forma dinámica. Este proceso elimina la necesidad de crear directorios/ficheros cache para cada módulo. Todo módulo tiene una estructura básica de directorios y archivos, y la funcionalidad está dada por un sistema de bloques. Un módulo puede incorporar tanto bloques como el desarrollador necesita para lograr así la funcionalidad que desea. Entre los directorios fundamentales que conforman la estructura de un módulo se encuentra el directorio `admin`, `blocks`, `templates`, `class`, `include` etc, para una mejor familiarización con esta estructura ver **Anexo 5**.

Debido a la estructura modular de los sistemas manejadores de contenidos y la estructura particular que ofrece el xoops para desarrollar sus módulos, en esta sección se modelará el módulo según la funcionalidades que brinden al sistema y de acuerdo a la forma en que se ha organizado el contenido del trabajo, se deben presentar los modelos organizados por paquetes, que como se dijo antes responderán a una funcionalidad del sistema, de forma que pueda entenderse mejor la lógica del negocio.

Paquete 1 Perfil del QuipusNews.

Paquete 2 Perfil del Editor.

Sub-Paquete 2.1 Manejar Contenido Revistas.

Sub-Paquete 2.2 Manejar Contenido Artículos.

Sub-Paquete 2.3 Manejar Contenido Noticias.

Sub-Paquete 2.4 Manejar Contenido Secciones.

Paquete 3 Perfil del Usuario Anónimo.

Sub-Paquete 3.1 Mostrar Contenido Revista.

Sub-Paquete 3.2 Mostrar Contenido Noticias.

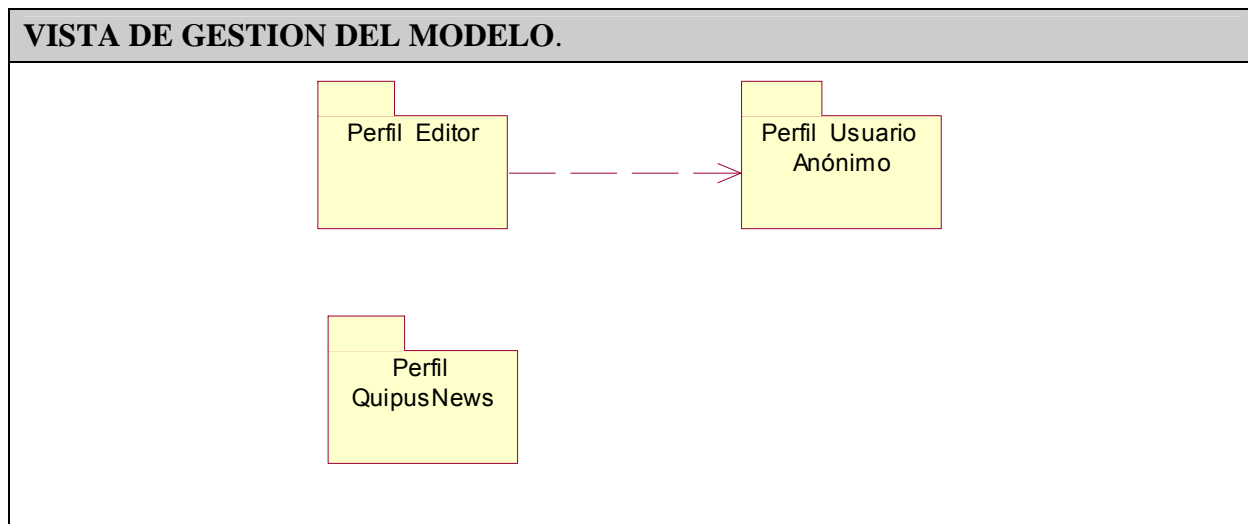
Sub-Paquete 3.3 Mostrar Colección Dossier.

Sub-Paquete 3.4 Mostrar Ediciones Anteriores.

Por la similitud en la funcionalidad de varios paquetes se representara los diagramas de Web de los paquetes más importantes, el resto se agrega a los anexos (ver **Anexo 6, Anexo 7 y Anexo 8**).

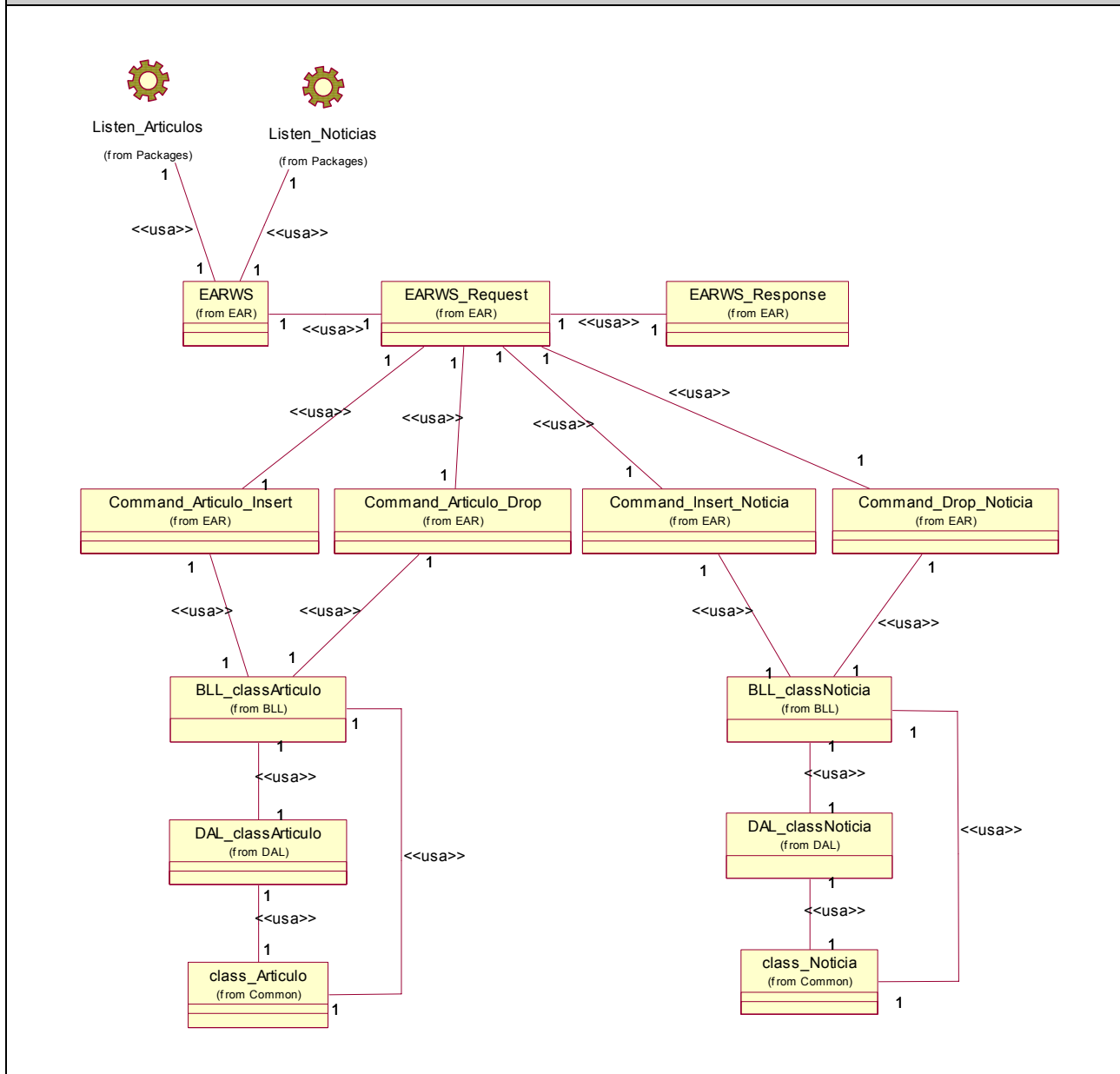
Para una mejor comprensión de los diagramas y en el modelo se especificara a continuación la nomenclatura utilizada en las clases.

Clases	Significado
DAL	Data Access Layer, representa una clase utilizada para el acceso a datos
BLL	Business Logic Layer, representa una clase que implementa reglas de negocio.
class	Representa una clase persistente
Form	Representa un formulario WEB.
Page	Representa una página HTML
XP	Este prefijo se usara para las clases que pertenezcan a Xoops pero que se hace necesario incluirlas en el modelo por su relación con el mismo.
admin	Representa las clases definidas con fines de administración.
inc	Representa las clases que brindan funciones útiles usadas.
Block	Representa un bloque del módulo.
EARWS	Representa las clases definidas para manipular el contenido recibido.
Command	Clases que se utilizan para invocar los métodos de insertar o eliminar el contenido recibido.

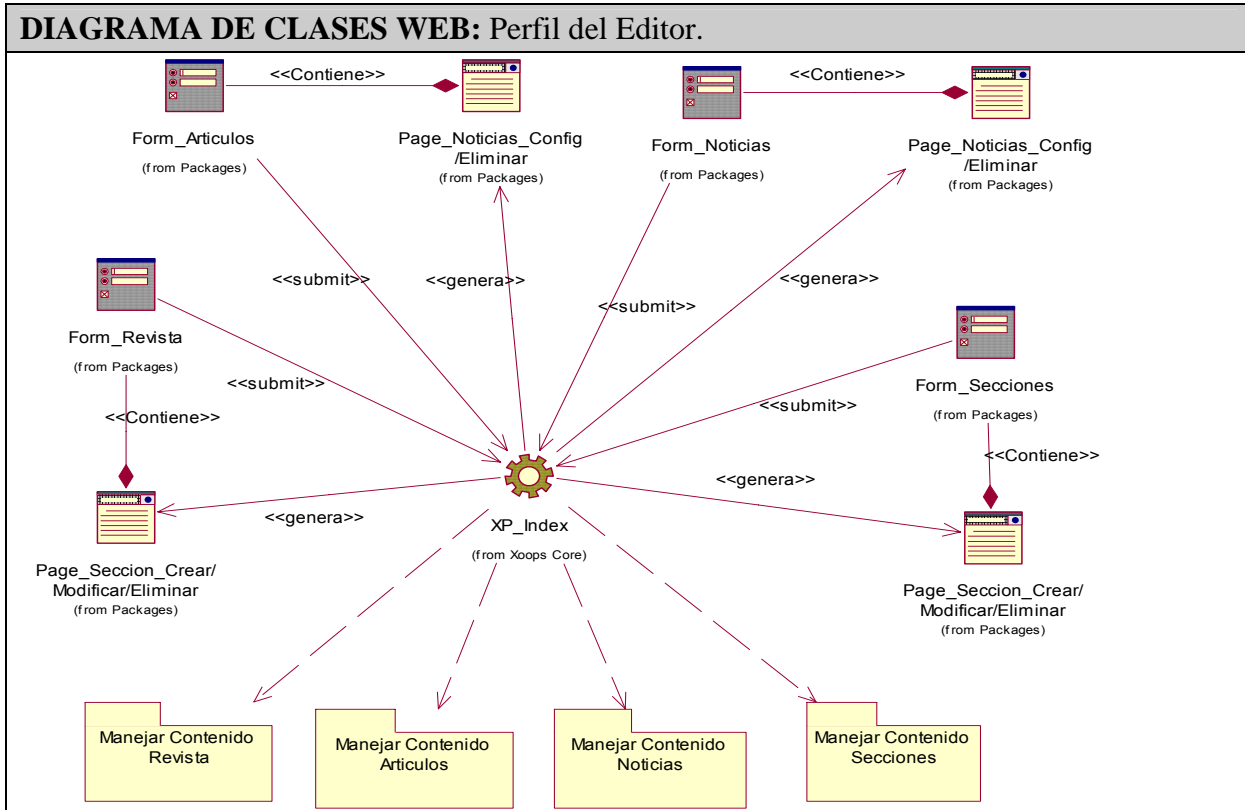


4.2.1 Paquete 1 Perfil de QuipusNews.

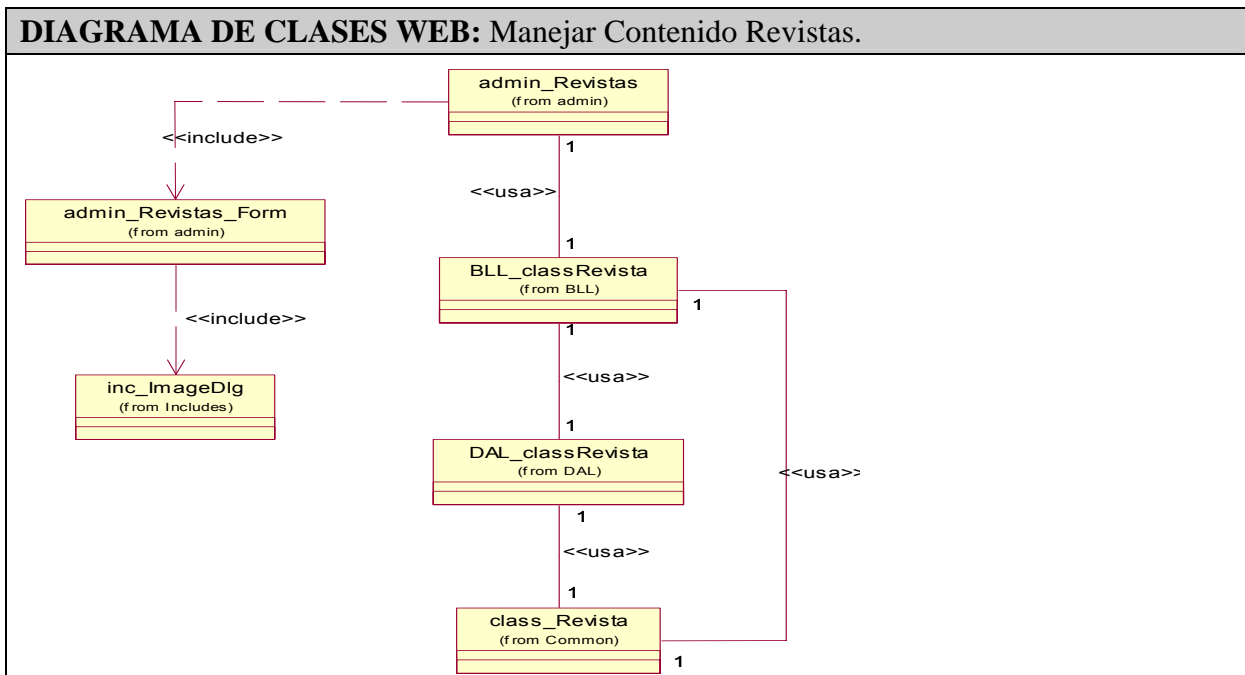
DIAGRAMA DE CLASES WEB: Recibir e Insertar contenido de artículos y noticias.



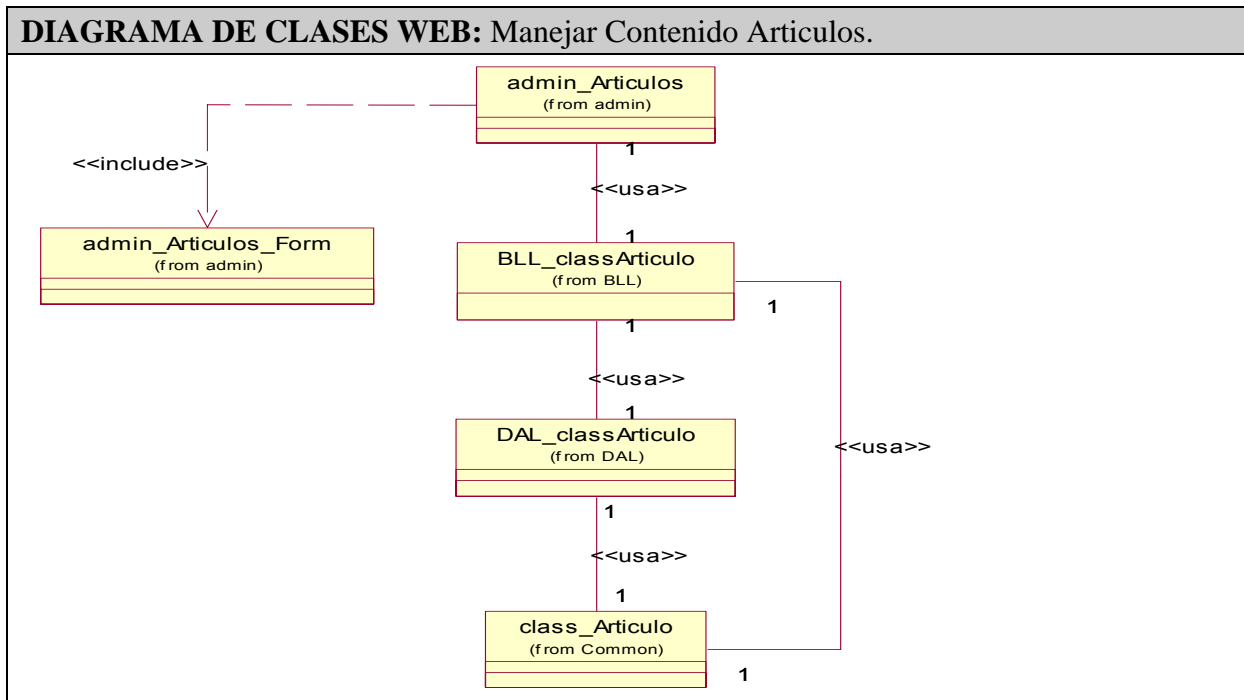
4.2.2 Paquete 2 Perfil del Editor.



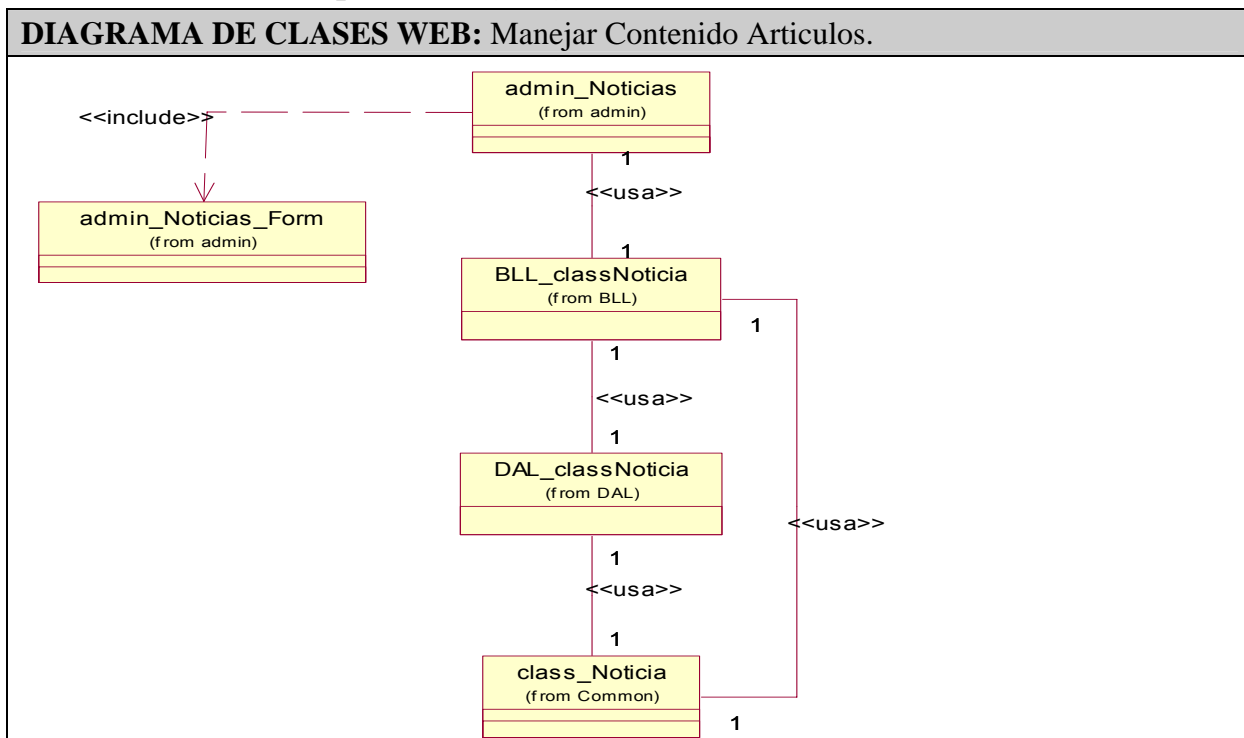
4.2.2.1 Sub-Paquete 2.1 Manejar Contenido de Revista.



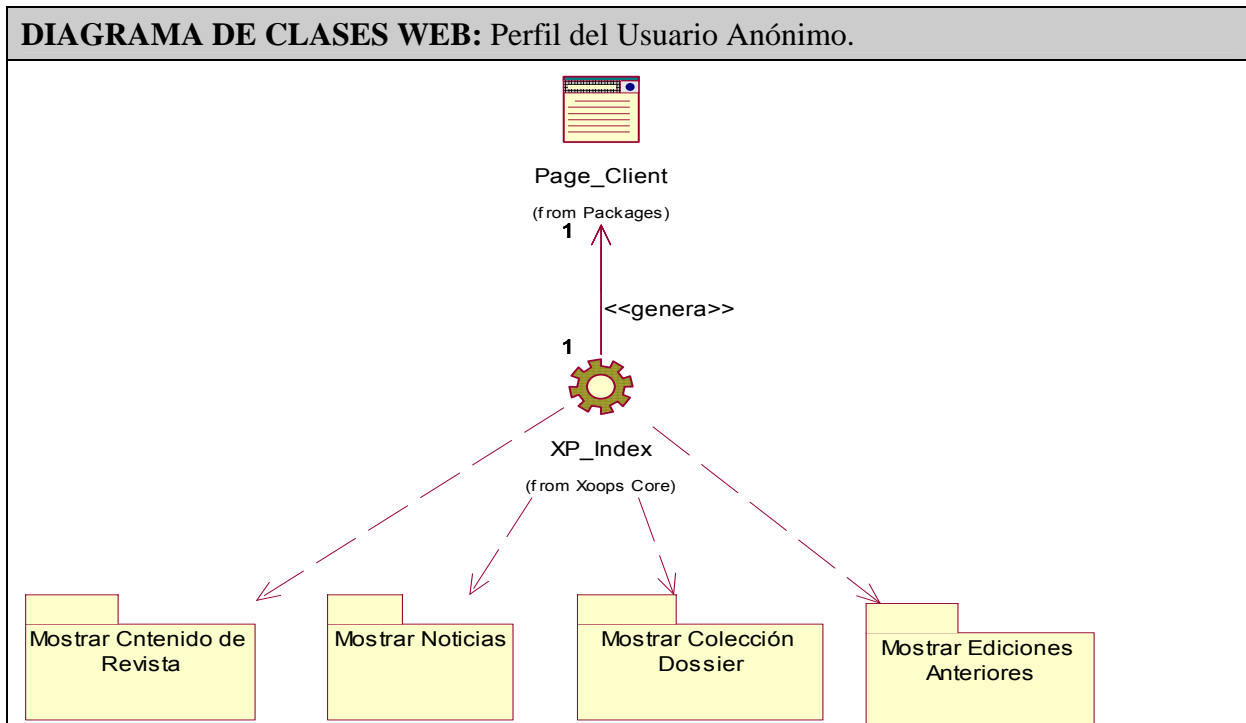
4.2.2.2 Sub-Paquete 2.2 Manejar Contenido de Artículos.



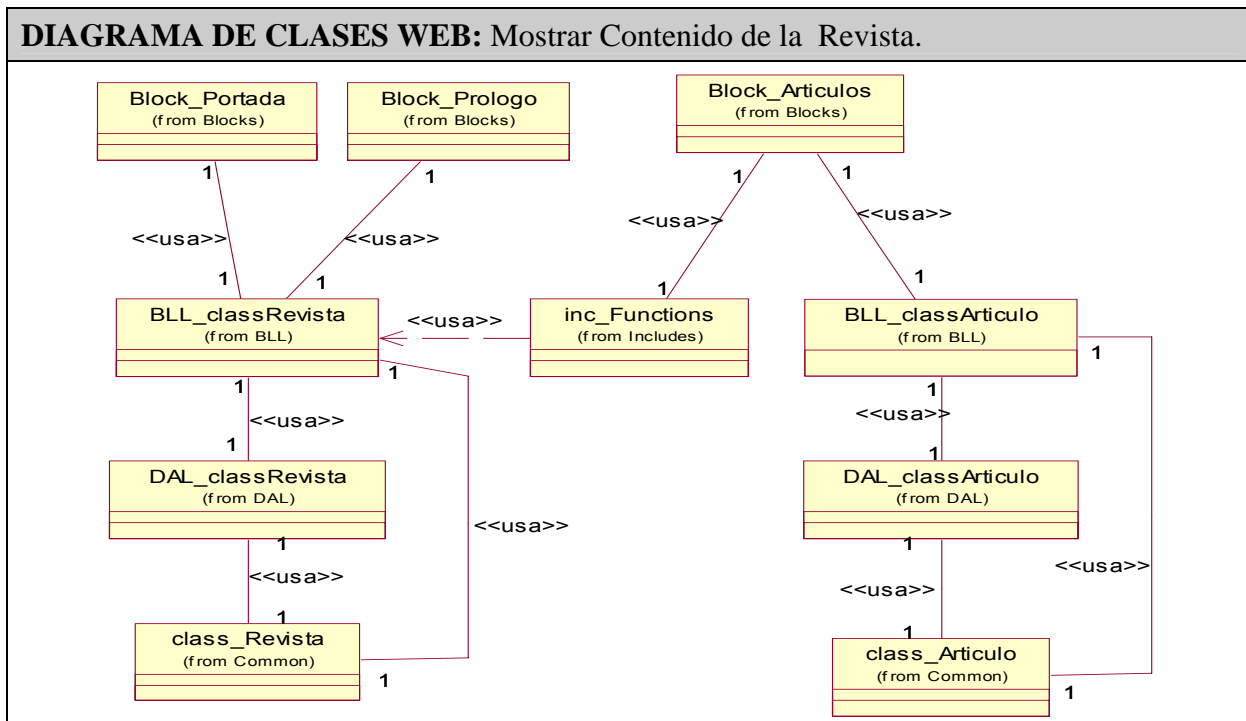
4.2.2.3 Sub-Paquete 2.3 Manejar Contenido de Noticias.



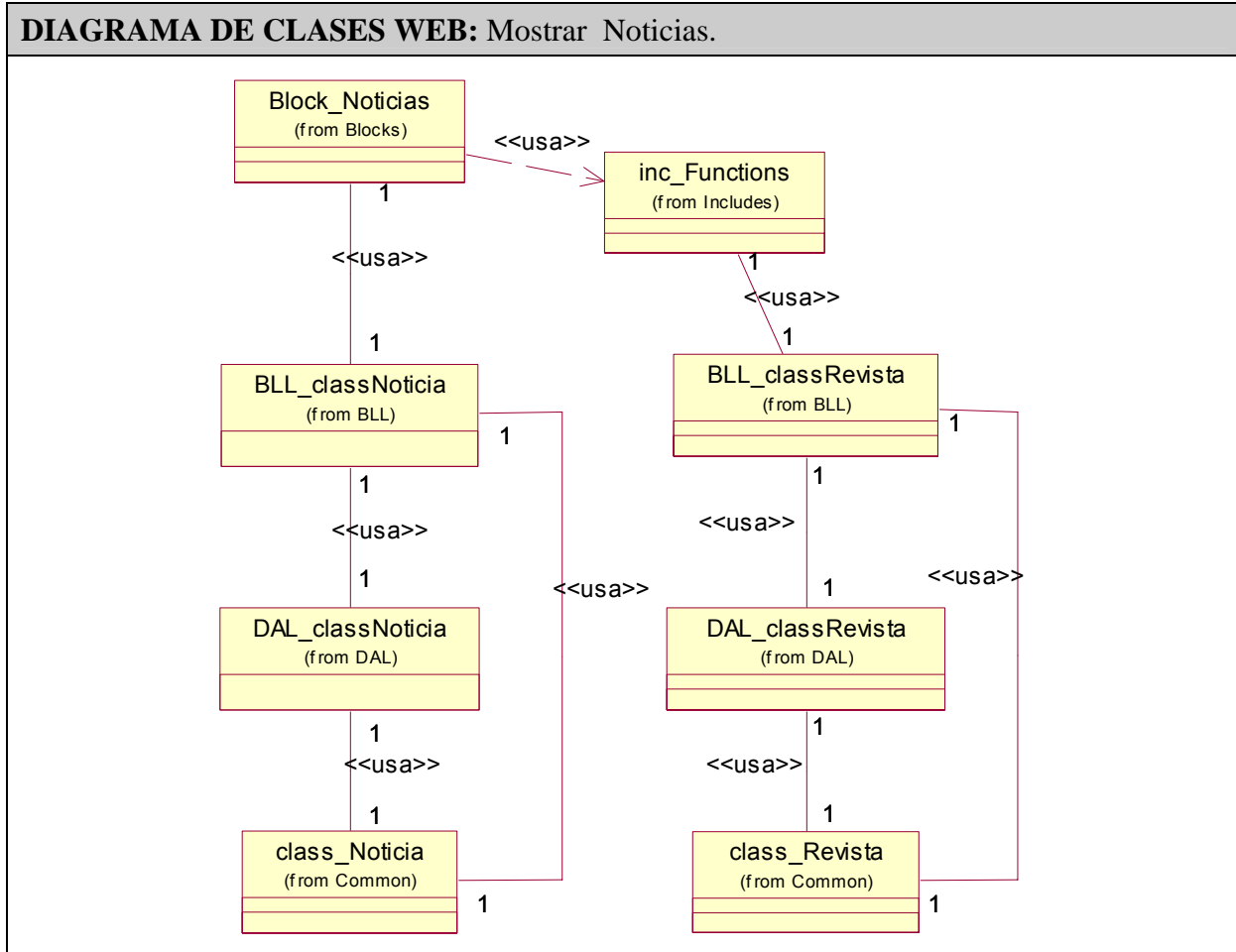
4.2.3 Paquete 3 Perfil del Usuario Anónimo.



4.2.3.1 Sub-Paquete 3.1 Mostrar Contenido de la Revista.



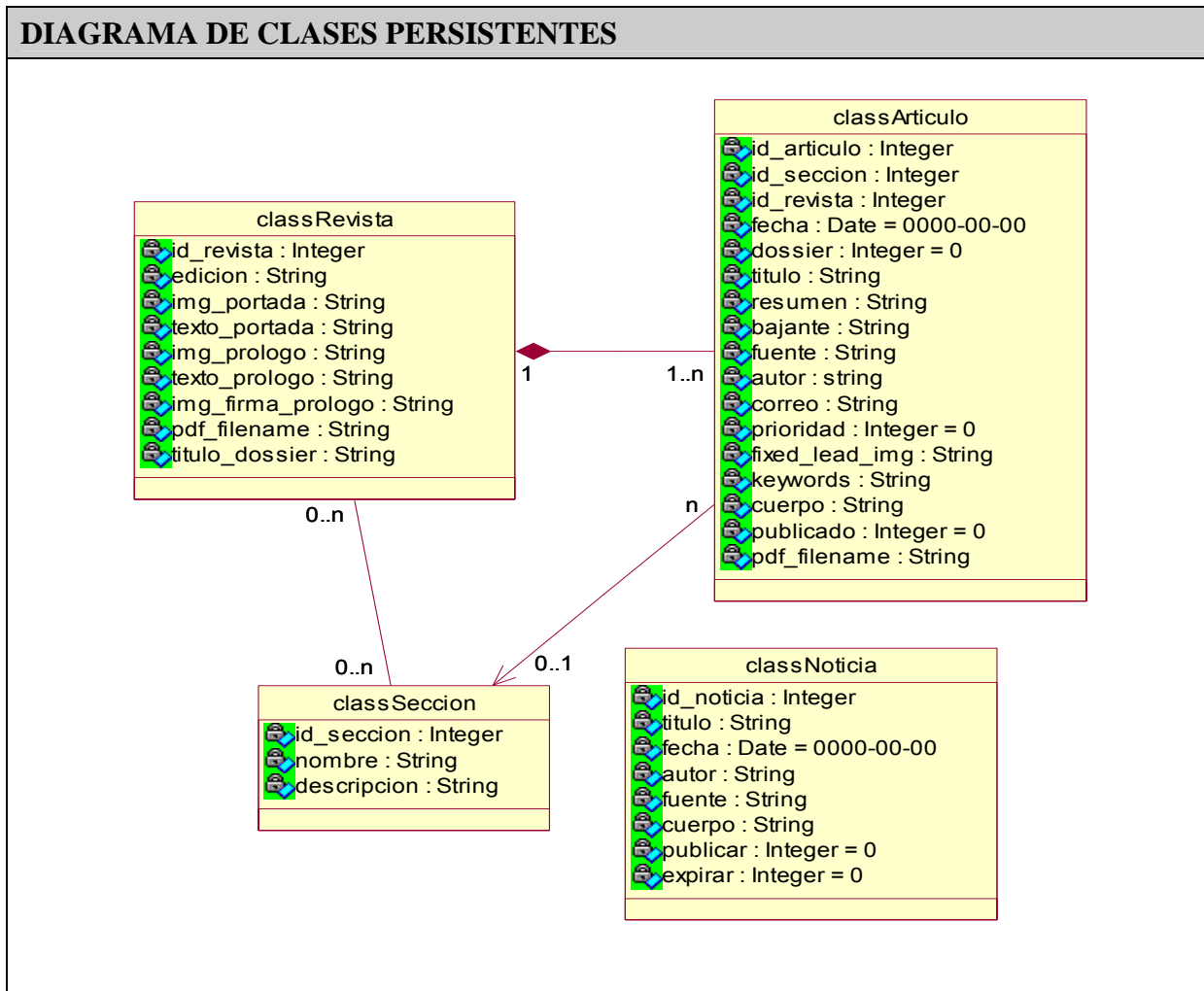
4.2.3.2 Sub-Paquete 3.2 Mostrar Noticias.



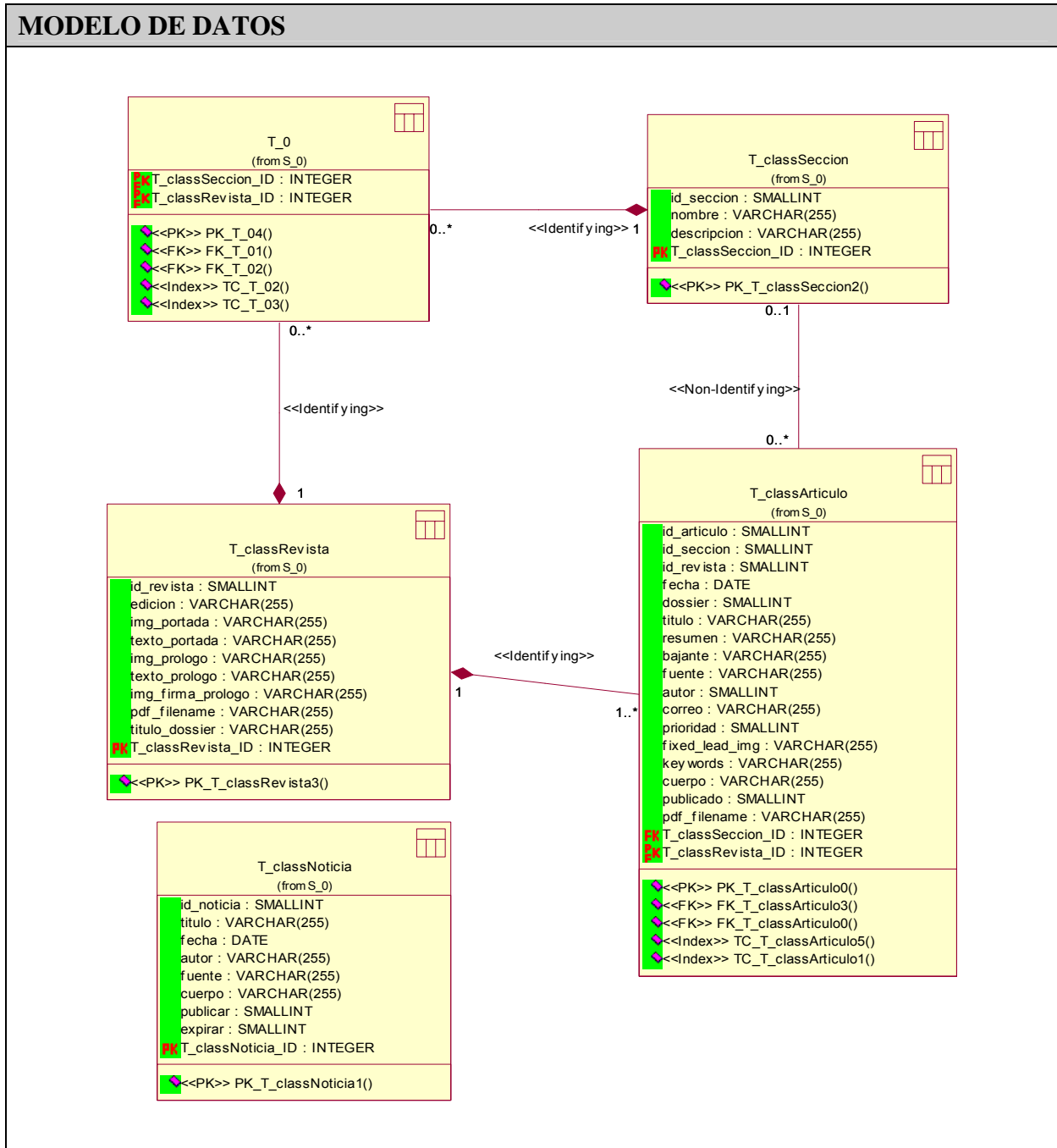
4.3 DISEÑO DE LA BASE DE DATOS.

A continuación se brinda el diagrama de clases persistentes y el modelo de datos que dan soporte al contenido manejado por el sistema. Es bueno aclarar que en el modelo de datos no se incluyen las entidades que son manejadas por el sistema de manejador de contenidos Xoops, por ejemplo los usuarios así como aquellas tablas que utilizan módulos usados en el portal, tal es el caso de los módulos de las Encuestas, la Galería de Imágenes y los Enlaces.

4.3.1 Clases persistentes.



4.3.2 Diagrama del modelo de datos.



4.4 PRINCIPIOS DE DISEÑO, MANEJO DE ERRORES Y ESTÁNDARES DE CODIFICACIÓN.

4.4.1 Principios de Diseño

En este caso el diseño de la aplicación se logra con la construcción de una plantilla para el Sistema de Gestión de Contenido Xoops. La 'piel' (skin) de un sitio XOOPS es controlado por un poderoso sistema de temas gráficos que permite al diseñador del sitio trabajar directamente en HTML.

El diseño, sea cual sea el objeto del mismo, tiene que basarse en el usuario, y en este caso estamos hablando de cualquier persona en el mundo; donde la gran mayoría no cuentan con una preparación en las cuestiones de la informática. Para ello, este sistema utiliza ciertos principios generales de diseño que garantizan la usabilidad en los diseños para las aplicaciones Web.

Principio de uso equiparable: donde las características de privacidad, garantía y seguridad estén igualmente disponibles para todos los usuarios, y que el diseño sea atractivo para todos los usuarios.

Principio de la flexibilidad: donde se ofrezcan posibilidades de elección en los métodos de uso, que facilite al usuario la exactitud y precisión, y se adapte al paso o ritmo del usuario.

Principio de la Información perceptible: donde se usen diferentes modos para presentar de manera redundante la información esencial (gráfica y verbal), se proporcione contraste suficiente entre la información esencial y sus alrededores, se amplíe la legibilidad de la información esencial, y que diferencie los elementos en formas que puedan ser descritas (por ejemplo, para las funciones de catalogación).

Principio de tolerancia al error: donde se dispongan los elementos para minimizar los riesgos y errores, por ejemplo utilizando elementos comunes; y los elementos peligrosos eliminados, aislados o tapados, que se proporcionen advertencias sobre

peligros y errores. Hay que posibilitar el descubrimiento interactivo y el aprendizaje ensayo-error, y posibilitar la reversibilidad y la recuperabilidad de las acciones.

Principio de esfuerzo de acceso y uso: que minimicen las acciones repetitivas, y que proporcione una línea de visión clara hacia los elementos importantes tanto para un usuario sentado como de pie.

4.4.2 Manejo de Errores.

El manejo de los errores esta a cargo del sistema de manejo de errores que usa Xoops. Por ejemplo cada vez que ocurre alguna excepción el xoops redirecciona al cliente a una pagina donde envía un mensaje reportando donde ocurrió el error y que lo ocasiono *Fig. 1*.

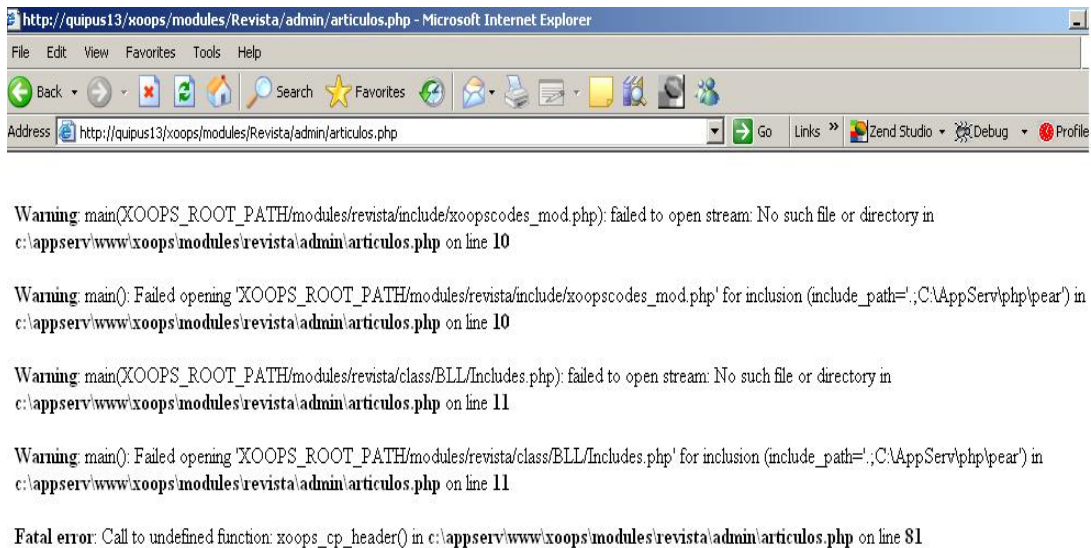


Fig.1 Página de Reporte de Error.

También al igual que en otros CMS se emiten mensajes de confirmación a la hora de hacer operaciones de eliminación *Fig.2*.

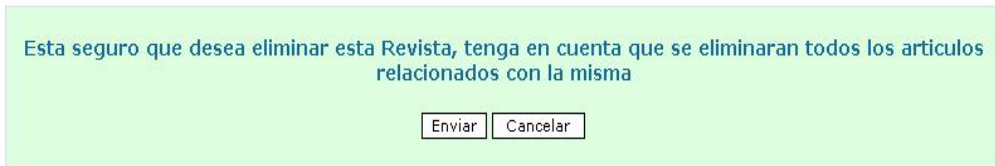


Fig.2 Confirmación de borrado.

Otros mensajes de alertas son emitidos por el xoops a los diferentes usuarios a la hora de tratar de acceder al lugares en el portal donde tiene el permiso restringido Fig.3, también emite avisos a la hora de chequear las credenciales de los usuarios, Fig.4 y Fig.5.

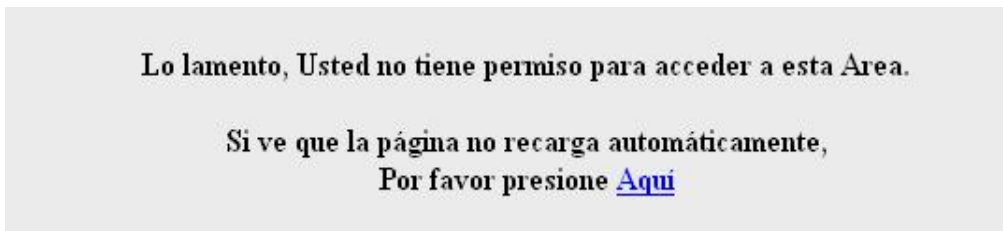


Fig.3 Acceso a un área restringida

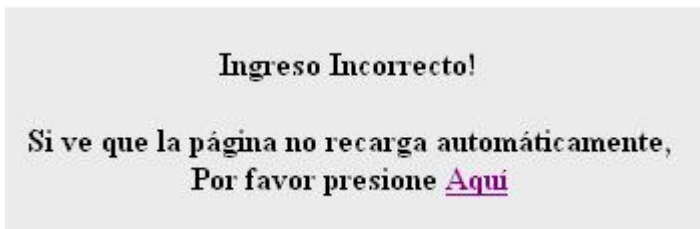


Fig.4 Aviso de Credenciales Incorrectas.

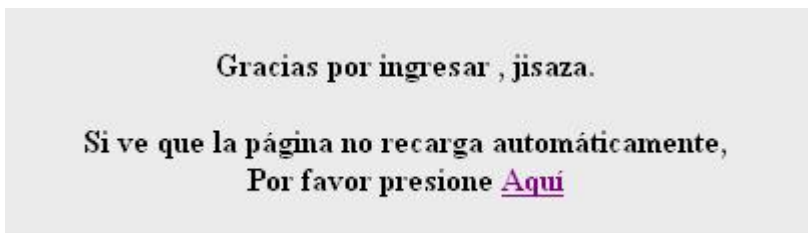


Fig.5 Aviso de Credenciales Correctas.

4.4.3 Estándares de codificación.

Para obtener un código más claro y comprensible, y facilitar el mantenimiento del software hemos seguido ciertos estilos o estándares a la hora de escribir el código del software. A continuación se muestra una serie de normas que se tuvieron en cuenta.

Bloques de instrucciones. Colocar el corchete de apertura {, al final de la instrucción y alinear el corchete de cierre} con la línea de instrucción. El código dentro del bloque se indento con el tabulador, para permitir mayor legibilidad se configuro el editor para que despliegue el tabulador con una longitud de cuatro espacios (Fig. 6).

```
267 function Eliminar_Articulo( $id_articulo ){
268     $Bll_Obj = new BLLArticulo();
269     if ( $Bll_Obj->Delete( $id_articulo ) ){
270         redirect_header("articulos.php", 1, "Artículo Eliminado");
271     } else {
272         redirect_header("articulos.php", 1, "Artículo no Eliminado");
273     }
274 }
```

Fig.6

Comparaciones Seguras. Poner siempre los valores constantes a la izquierda de la comparación "if (6 == \$variable)", con esto garantizará la generación de un error cuando por error escriba '=' y no '=='. (Fig. 7)

```
295
296 function Set_Img_Estado( $id_articulo ){
297     $Bll_Obj = new BLLArticulo();
298     if (0 == $Bll_Obj->Get_Estado( $id_articulo ) ){
299         return '';
300     } else {
301         return '';
302     }
303 }
```

Fig.7

Nombres.

A continuación se define la manera en la que se asignaron nombres a clases, métodos, funciones, variables y parámetros.

Clases. Primera letra de cada palabra en mayúscula sin separadores (Fig. 8).

Atributos de una clase y variables. Los atributos se escribieron con letra minúscula usando guión inferior para separar cada palabra (Fig. 8).

Métodos y funciones. La primera letra en minúscula y la primera letra de las palabras siguientes en mayúscula, usando guión inferior para separar las palabras (Fig. 8).

Argumentos o parámetros. Los argumentos pasados a una función o método inician con la letra p y la primera letra de las palabras siguientes en minúscula (Fig. 8).

```

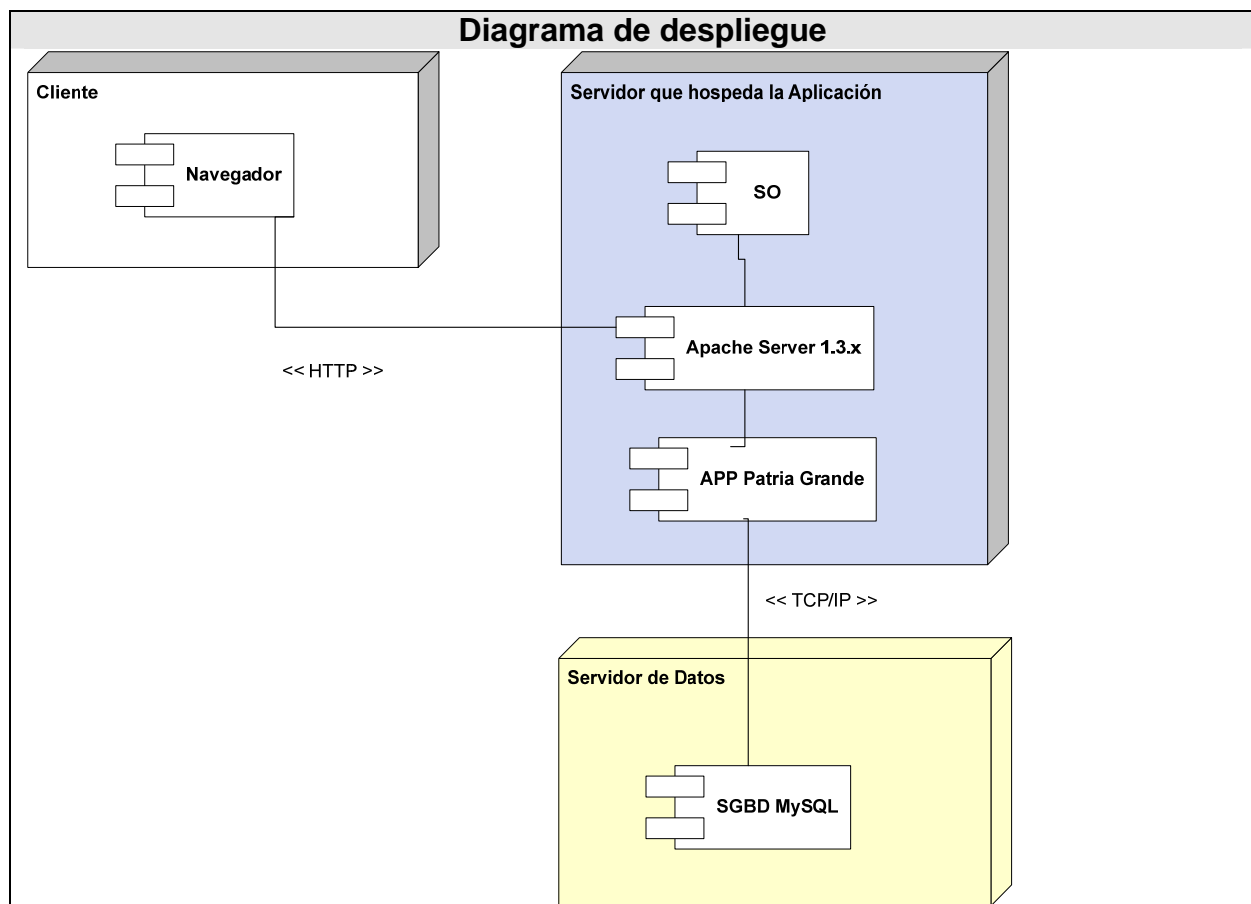
1  <?php
2  class Seccion{
3      var $id_seccion;
4      var $nombre;
5      var $descripcion;
6      function Seccion( $pid_seccion, $pnombre, $pdescripcion ){
7          $this->id_seccion = $pid_seccion;
8          $this->nombre     = $pnombre;
9          $this->descripcion = $pdescripcion;
10     }
11     function get_Id_Seccion(){
12         return $this->id_seccion;
13     }
14     function get_Nombre(){
15         return $this->nombre;
16     }
17     function get_Descripcion(){
18         return $this->descripcion;
19     }
20 }
21 ?>

```

Fig. 8

4.6 MODELO DE DESPLIEGUE.

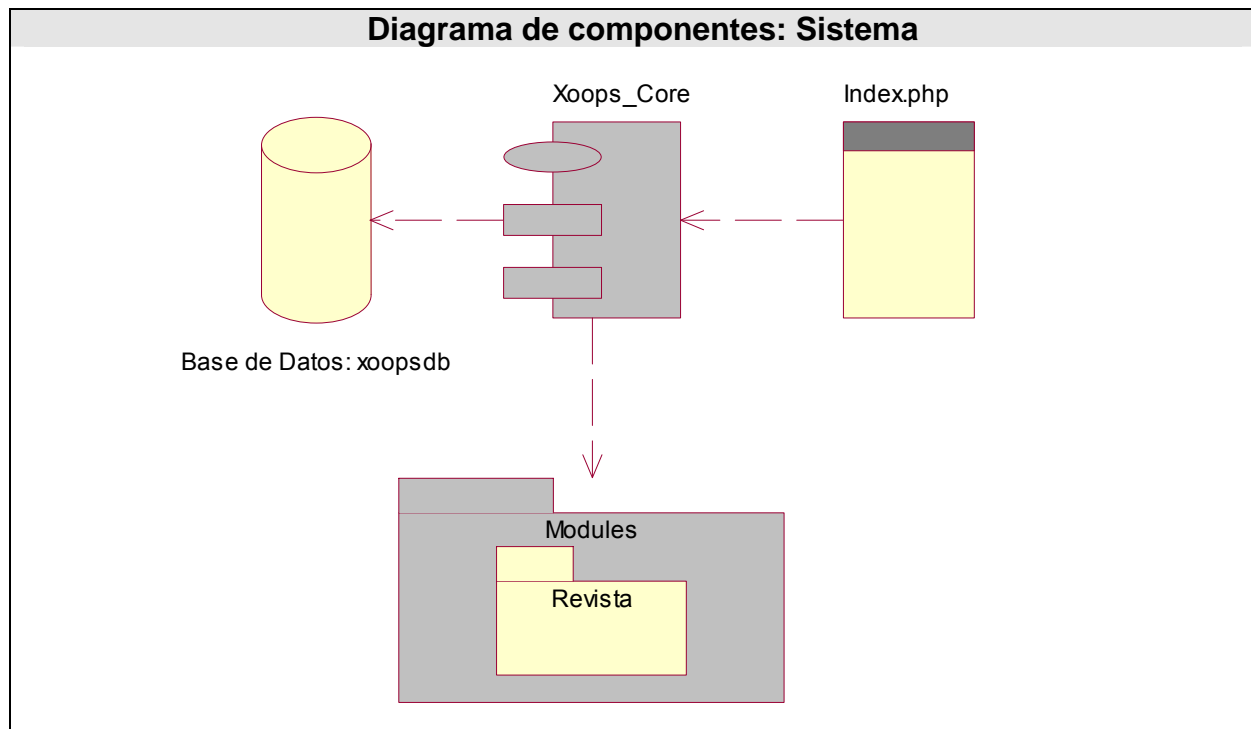
Un diagrama de despliegue muestra la configuración de los nodos que participan en la ejecución y de los componentes que residen en ellos y mediante el mismo podemos ver como se encuentran relacionados físicamente los componentes de la aplicación.

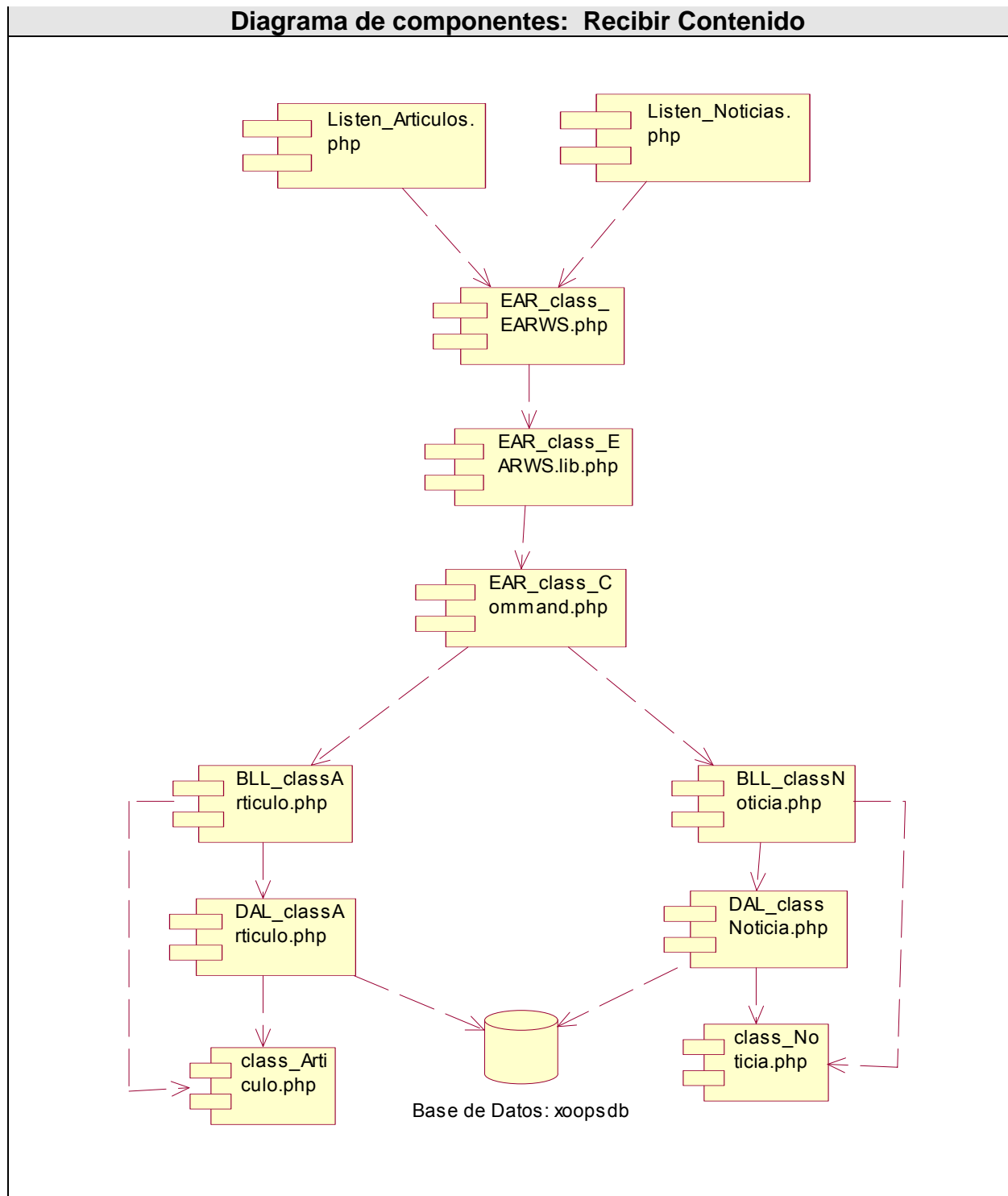


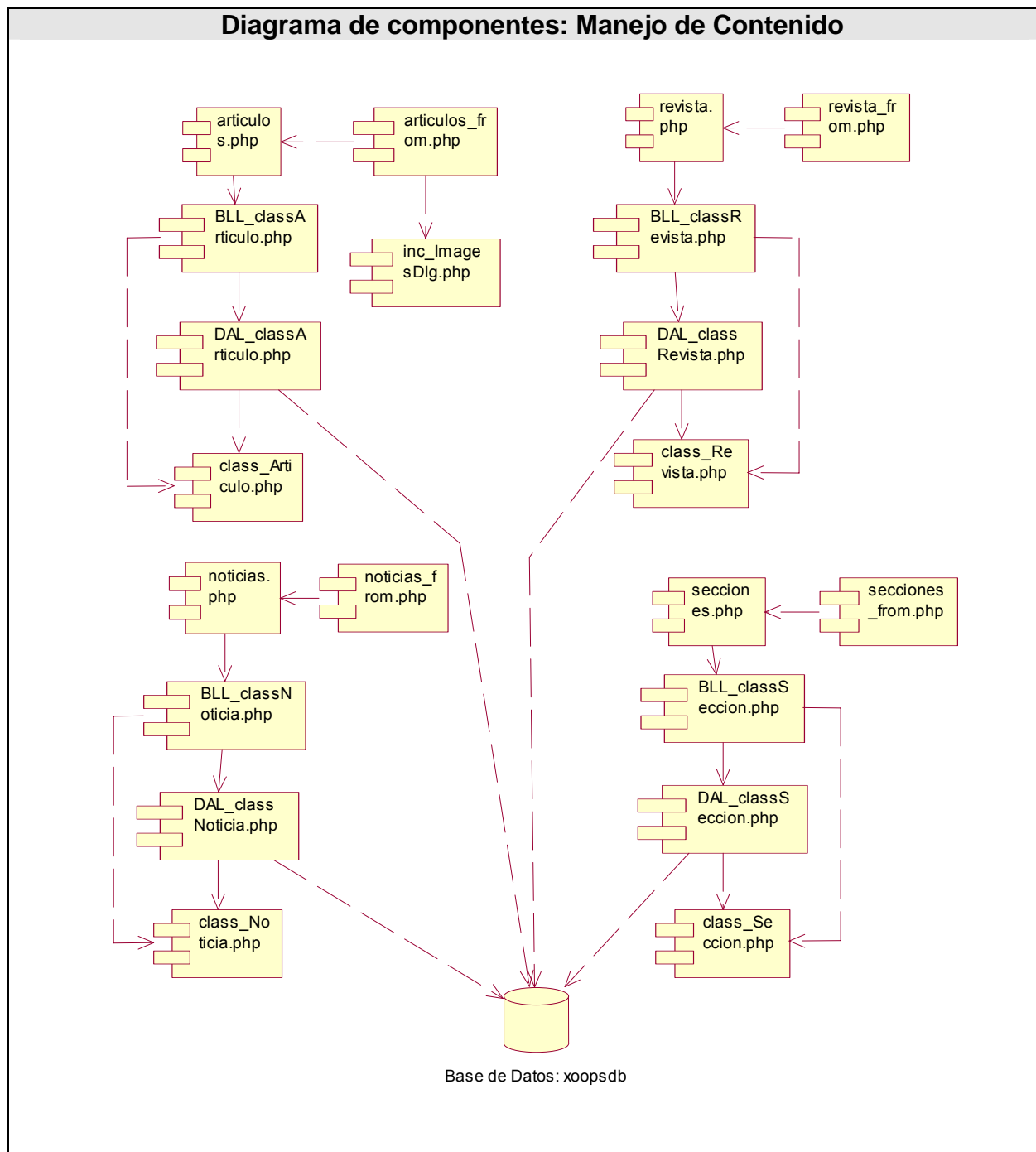
4.7 MODELO DE IMPLEMENTACIÓN.

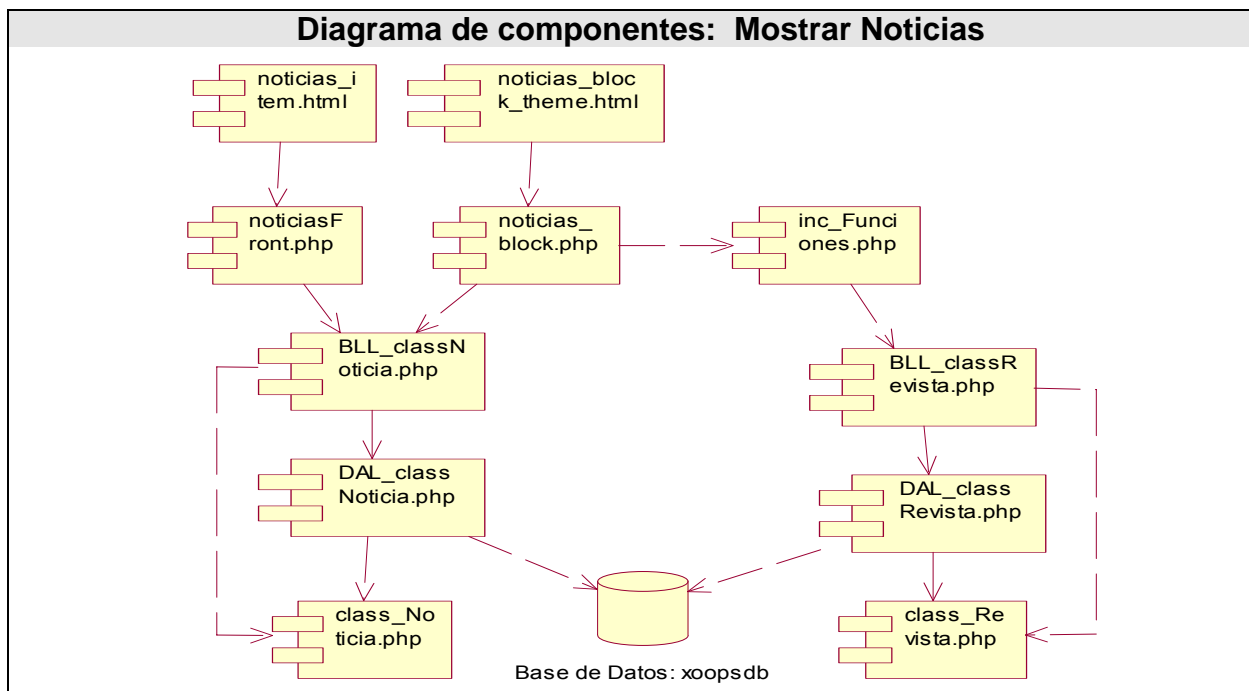
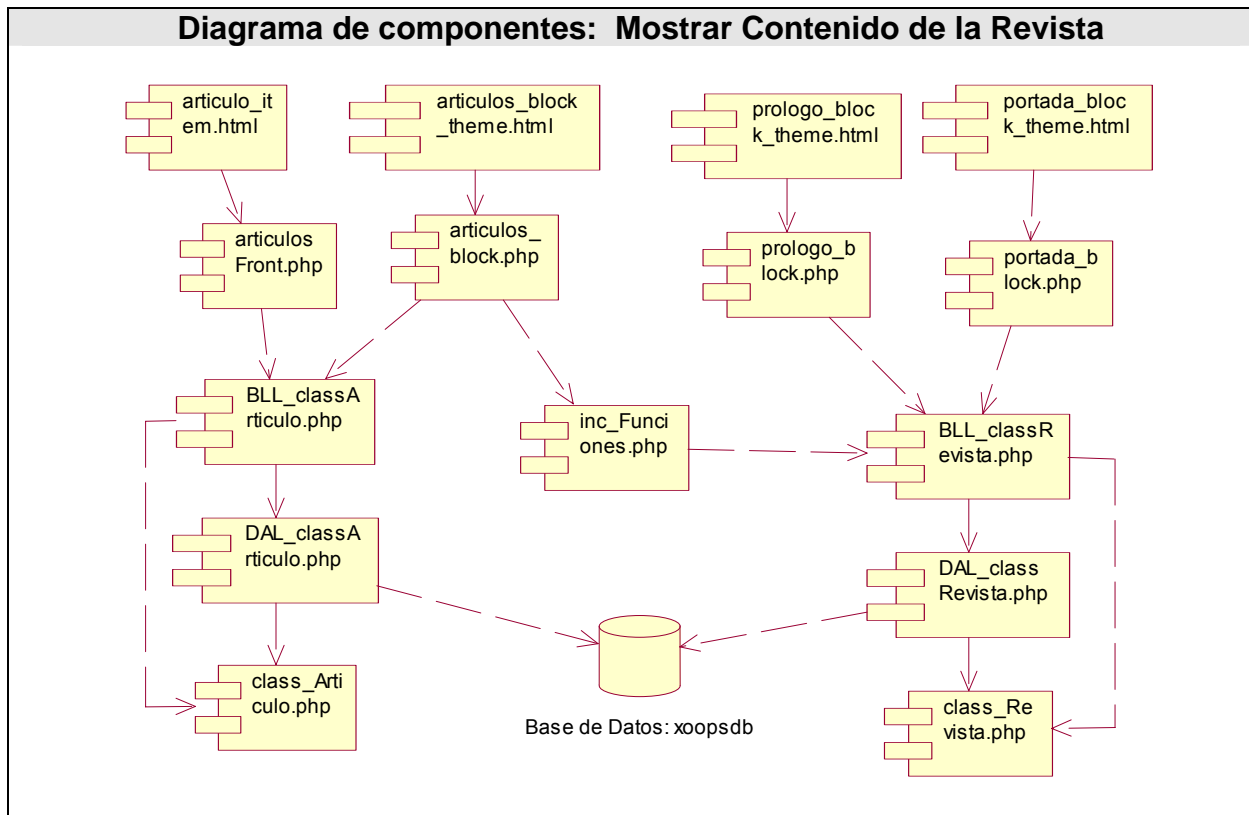
A continuación presentamos el diagrama de componentes, el cual contribuirá a un mejor entendimiento del modelo de implementación, dado lo extenso que podría ser

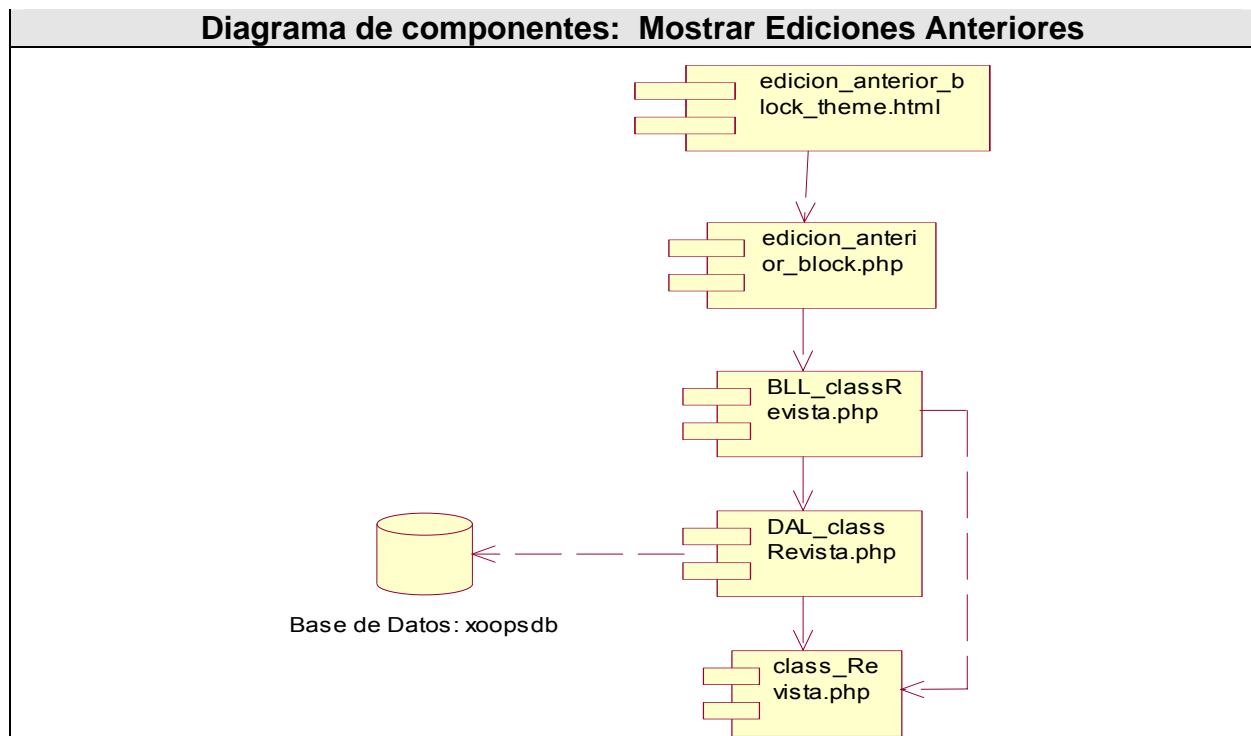
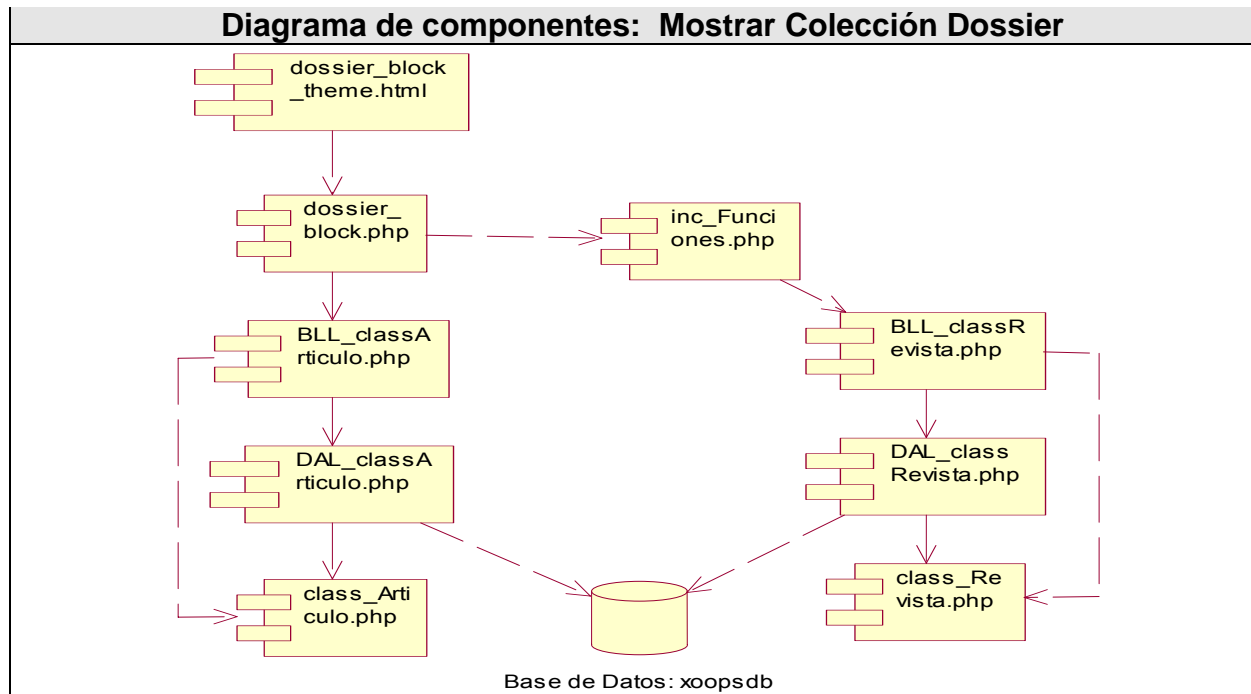
hacer toda la representación en un solo diagrama se decidió dividirlo en diagramas más sencillos para lograr así más claridad en las representaciones, por ejemplo los diagramas de mostrar contenido. Con ellos se representan los componentes lógicos de la aplicación así como las relaciones que existen entre ellos. Tengamos en cuenta que los paquetes de color diferente corresponden al Sistema de Gestión de Contenidos Xoops, los cuales no están modelados además no se representan todos los paquetes que utiliza el Xoops, solo aquellos más importantes y en los cuales se programo en este caso el módulo Revista. Otra aclaración es que el paquete **Modules** contiene muchos otros módulos pero solo se representara el módulo que se desarrollo para dar solución a la propuesta de solución.











Componente	Propósito	Contenido	Interfaces
Index.php	Es el fichero que da inicio la aplicación.	-	-
Xoops_Core	Aquí se en cuenta el API del cms Xoops.	-	-
Base de Datos:xoops db	Dar soporte a la persistencia del os datos.		
index.php	Brindar acceso a un índice con las opciones administrativas del módulo.	-	-
menú.php	Lo mismo que el anterior.	-	-
revista.php	Administrar los datos generales de las revistas, como, prestación editorial, imágenes de portada, etc.	Contiene los métodos para generar las interfaces de usuario para la administración y configuración de los elementos de la revista.	Listar_Revistas() Adicionar_Revista_Form() Adicionar_Datos_Revista() Revista_Editar() Listar_Secciones_CheckBox() Revista_Actualizar() Eliminar_Revista()
articulos.php	Administrar los artículos, configúralos etc.	Lo mismo que el anterior pero para los artículos.	Listar_Articulos() Editar_Articulo() Actualizar_Articulo() Listar_Revistas_Combos() Listar_Secciones_Combos() Dossier_Checked() Titular_Checked() Revista_Info() Eliminar_Articulo()
noticias.php	Administrar las noticias, configúralas etc.	Lo mismo que el anterior pero para las noticias.	Listar_Noticias() Editar_Noticia() Actualizar_Noticia() Eliminar_Noticia()
secciones.php	Administrar las secciones,	Lo mismo que el anterior pero para las secciones.	Listar_Secciones() Adicionar_Secciones_F

	configúralas etc.		orm() Adicionar_Datos_Seccion() Seccion_Editar() Secciones_Actualizar() Eliminar_Seccion()
revista_form.php	Generar código html para los formularios.	-	-
noticias_form.php	Generar código html para los formularios.	-	-
articulos_form.php	Generar código html para los formularios.	=	-
secciones_form.php	Generar código html para los formularios.	-	-
portada_bloque.php	Cargar la portada de la revista publicada.	Contiene el método que utiliza xoops para cargar al bloque la portada de la revista.	Cargar_Portada()
prologo_bloque.php	Cargar el prologo o presentación editorial de la revista publicada.	Contiene el método que utiliza xoops para cargar al bloque la presentación editorial o prologo de la revista.	Cargar_Prologo()
articulos_bloque.php	Cargar los artículos pertenecientes a la revista publicada u otra seleccionada.	Contiene el método utilizado por xoops para cargar al bloque la lista de los artículos de la revista publicada.	Cargar_Articulos()
noticias_bloque.php	Cargar las noticias publicadas en el portal.	Contiene el método utilizado por xoops para cargar al bloque la lista de las noticias publicada en el portal.	Cargar_Noticias()
dossier_bloque.php	Cargar la colección de artículos dossier.	Contiene el método utilizado por xoops para cargar al bloque la lista con la colección de artículos dossier.	Seleccion_Dossier()
seleccion_bloque.php	Cargar las ediciones de las revistas de los meses anteriores	Contiene el método utilizado por xoops para cargar al bloque la lista con las ediciones de las revistas de los meses anteriores.	Seleccion()
class_BLLArticulo.php	Lógica de negocio.	BLL_classArticulo	Insertar(), Eliminar(), Actualizar()

			Seleccionar_Todos(), Seleccionar_Uno(),Publicar(),Get_Estado(), Set_Config(), Select_Articulos_Revista(),Select_Articulos_por_Seccion(),Select_Dossier().
class_BLLNoticia.php	Lógica de negocio.	BLL_classNoticia	Insertar(), Eliminar(), Actualizar(), Seleccionar_Todo(), Seleccionar_Uno(), Set_Estados_Publicacion().
class_BLLRevista.php	Lógica de negocio.	BLL_classRevista	Insertar(), Eliminar(), Actualizar(), Seleccionar_Todo(), Seleccionar_Uno(), los otros métodos son de selección por diferentes criterios.
class_BLLSeccion.php	Lógica de negocio.	BLL_classSeccion	Insertar(), Eliminar(), Actualizar(), Seleccionar_Todo(), Seleccionar_Uno().
class_DALArticulo.php	Acceso a datos.	DAL_classArticulo	Los mismos que BLL_classArticulo.
class_DALNoticia.php	Acceso a datos.	DAL_classNoticia	Los mismos que BLL_classNoticia.
class_DALRevista.php	Acceso a datos.	DAL_classRevista	Los mismos que BLL_classRevista.
class_DALSeccion.php	Acceso a datos.	DAL_classSeccion	Los mismos que BLL_classSeccion.
classArticulo.php	Clase persistente.	classArticulo	-
classNoticia.php	Clase persistente.	classNoticia	-
classRevista.php	Clase persistente.	classRevista	-
classSeccion.php	Clase persistente.	classSeccion	-
class_Command.php	Adicionar y eliminar los artículos y noticias a la base de datos.	Command_Articulo_Insertar Command_Noticia_Insertar Command_Articulo_Eliminar Command_Noticia_Eliminar	Los metodos de Adicionar y Eliminar.
class_EARWS.lib.php	Procesar datos recibidos.	EARWS_Request EARWS_Response	RunRequest(), Chequear_Autenticacion

			n(), Parsear_Daos(), Prepare_Response(), Send_Response().
class_EAR WS.php	Procesar datos recibidos.	EARWS	Create_Request().
Listen_Articulos.php	Recibir datos de artículos.	-	-
Listen_Noticias.php	Recibir datos de noticias.	-	-
articulosFront.php	Mostrar el contenido completo de un artículo seleccionado.	Contiene el método para mostrar un artículo seleccionado.	-
noticiasFront.php	Mostrar el contenido completo de una noticia seleccionada.	Contiene el método para mostrar una noticia seleccionada.	-

4.8 CONCLUSIONES.

En este capítulo se mostraron los pasos para llevar a cabo el proceso de implementación del sistema basándose en la estructura de los módulos desarrollados para xoops y los principios de la Programación Orientada a Objetos. Se representaron mediante diagramas de clases Web la lógica del negocio del sistema, se diseñaron y representaron mediante diagramas las clases persistentes que dieron lugar al diagrama de entidad-relación en el sistema gestos de base de datos que se utilizara en la propuesta. Por los que podemos decir que ha quedado plasmada completamente la propuesta que trae este trabajo.

Capítulo V.

ESTUDIO DE FACTIBILIDAD.

5.1 INTRODUCCIÓN.

"Para llevar a cabo un buen proyecto de desarrollo de software, debemos comprender el ámbito del trabajo a realizar, los recursos requeridos, las tareas a ejecutar, las referencias a tener en cuenta, el esfuerzo (COSTE) a emplear y la agenda a seguir." (R. Pressman)

Para intentar dar solución a estos problemas se han introducido en la Ingeniería del Software una serie de técnicas, utilizadas dentro de las tareas de planificación, que ayudan a planificar y controlar el esfuerzo y el tiempo necesario de desarrollo. Para realizar estas estimaciones utilizamos el modelo de COCOMO (*CO*nstructive *CO*st *MO*del).

5.2 PLANIFICACIÓN.

Nombre de la entrada externa	Cantidad de Ficheros	Cantidad de Elementos de datos	Clasificación (Simple, Media y compleja)
Crear Revista	1	9	Simple
Modificar Revista	2	11	Medio
Eliminar Revista	3	28	Complejo
Adicionar Artículos	1	17	Medio
Configurar Artículos	1	6	Simple
Eliminar Artículos	1	17	Medio
Adicionar Noticias	1	8	Simple
Configurar Noticia	1	2	Simple
Eliminar Noticias	1	8	Simple
Crear Secciones	1	3	Simple
Modificar Secciones	1	2	Simple
Eliminar Secciones	1	5	Simple

Tabla 1 Entradas Externas

Nombre de la salida externa	Cantidad de Ficheros	Cantidad de Elementos de datos	Clasificación (Simple, Media y compleja)
Listado de Revistas	1	2	Simple
Listado de Artículos	1	7	Simple
Listado de Noticias	1	6	Simple
Listado de Secciones	1	3	Simple
Mostrar revista del mes	3	10	Medio
Mostrar Colección Dossier	1	1	Simple
Mostrar Ediciones Anteriores	1	1	Simple

Tabla 2 Salidas Externas

Nombre de la petición	Cantidad de Ficheros	Cantidad de Elementos de datos	de	Clasificación (Simple, Media y compleja)
Visualizar Contenido Artículos	1	3		Simple
Visualizar Contenido Noticias	1	2		Simple

Tabla 3. Peticiones.

Nombre del fichero interno	Cantidad de records	Cantidad de Elementos de datos	Clasificación (Simple, Media y compleja)
xoops_revista	1	9	Simple
xoops_articulos	1	17	Simple
xoops_noticias	1	8	Simple
xoops_sección	1	3	Simple
xoops_revista_sección	1	2	Simple

Tabla 4. Ficheros Internos.

Elementos	Simples		Medios		Complejos		Subtotal de puntos de función
	No	X Peso	No	X Peso	No	X Peso	
Ficheros lógicos internos	5	7	0	10	0	15	35
Entradas externas	8	3	3	4	1	6	42
Salidas externas	6	4	1	5	0	7	29
Peticiones	2	3	0	4	0	6	6
Total							115

Tabla 5. Puntos de Función desajustados

5.3 COSTOS.

Características	Valor
Puntos de función desajustados	115
Lenguaje	PHP (85 %) JavaScript (3 %) HTML (12 %)
Instrucciones fuentes por puntos de función	(21) (56) (43)
Instrucciones fuentes por lenguaje (miles de instrucciones)	(2,0553) (0,1932) (0,5934)
Instrucciones fuentes (miles de instrucciones)	2,8419

Tabla 6. Cantidad de Instrucciones Fuentes.

Cálculo de:	Valor	Justificación
Esfuerzo	4,18	$\Pi EM_i \approx 0,5$ $E \approx 1$ $\sum SF_i = 20,04$ $PREC = 6,2$ $FLEX = 5,07$ $RESL = 5,65$ $TEAM = 0$ $PMAT = 3,12$
Tiempo de desarrollo	5,25 meses	$PM \approx 4,18$ $F \approx 0,32$
Cantidad de personas	1	Se cuenta con 1 persona para la realización del sistema.
Costo	940,5	$C = CHM * PM$ $CHM = 1 * 225 = 400$ $PM \approx 4,18$
Salario medio	225	
RCPX	0,83	$RELY = NOMINAL$ $DOCU = NOMINAL$ $CPLX = NOMINAL$

DATA = PEQUEÑO		
RUSE	1,00	Existe concordancia entre la documentación y las necesidades del ciclo de vida.
PDIF	1,00	TIME = 50% STOR = 50% PVOL = 50%.
PREX	1,00	APEX = NOMINAL PLEX NOMINAL LTEX = NOMINAL
FCIL	0,73	TOOL = MUY ALTO SITE = NOMINAL.
SCED	1,00	SCED = 130 %
PERS	0,83	ACAP = ALTO PCAP = ALTO PCON = ALTO

Tabla 7. Cálculos Finales de La Estimación de Costos y Esfuerzos.

5.4 BENEFICIOS TANGIBLES E INTANGIBLES.

El Portal de la revista Patria Grande es un producto con fines gubernamentales. Su principal objetivo es la publicación de información sobre la integración cubano-venezolana, la Alternativa Bolivariana para las Américas (ALBA) y el proceso revolucionario que encabeza el Presidente Hugo Chávez.

Los beneficios inmediatos son:

1. Acceso rápido y directo a la información publicada.
2. Permite la constante actualización del contenido, por parte de los editores, sin necesidad de que estos tengan conocimientos técnicos de HTML y desarrollo de aplicaciones Web.

3. Multiplicar el mensaje de la revista y hacerlo accesible a internautas en cualquier parte del mundo.
4. Utilizar las posibilidades de espacio, inmediatez y alcance de la web para convertir el sitio en un medio de análisis e información.

5.5 ANÁLISIS DE COSTOS Y BENEFICIOS.

El desarrollo de este sistema no supone grandes gastos de recursos, ni tampoco de tiempo. Las tecnologías y herramientas utilizadas para el desarrollo del mismo son totalmente libres, por lo que no hay que incurrir en gastos por pago de licencias de uso. El sistema es completamente portable, por lo que un cambio de plataforma es viable y factible, es completamente escalable gracias a la estructura basada en capas de los procesos del negocio que se diseñaron.

CONCLUSIONES.

La información es un elemento fundamental para el desarrollo, con el decursar de los años, la gestión de la información ocupa cada vez más un espacio mayor en el desarrollo de los países a escala mundial. Debemos gestionar y hacer el mejor uso posible de esa información, la cual debe ser relevante para alguien en un momento determinado, de lo contrario de poco nos servirá. Cada día son más y más las personas que pueden acceder a la Internet desde cualquier parte del mundo.

En los últimos años, países como Venezuela han combatido la influencia imperialista neoliberal, llevando a cabo los ideales revolucionarios de todos aquellos hombres que lucharon por la libertad de los pueblos de América, aquellos hombres que lucharon por una América libre.

Debido al conjunto de transformaciones sociales llevadas a cabo en el hermano país surge este proyecto, con el fin de insertar los programas sociales venezolanos y la integración Cuba-Venezuela en esta carrera de la información, donde las grandes compañías privadas de comunicación de ese país se encargan de tergiversar y manipular la verdad acerca de la realidad venezolana.

Con este trabajo se presenta una aplicación Web que permite gestionar más fácilmente el contenido de la revista así como facilitar la búsqueda de información a través de Internet acerca de los programas sociales venezolanos, la integración Cuba-Venezuela, la Alternativa Bolivariana para las Américas y demás temas abordados en la revista.

El sistema se desarrolló siguiendo la metodología RUP, y se utilizaron representaciones para la modelación de todas las fases del proyecto. Fue desarrollado por una sola persona en alrededor 4 meses sin costo monetario de ningún tipo. El sistema resultante provee un ambiente cómodo, fácil de entender, que cumple los estándares de diseño, codificación y desarrollo de módulos para los sistemas

manejadores de contenido, Xoops para este caso y además utiliza técnicas modernas de programación orientada a objetos.

Por tanto podemos concluir que los objetivos propuestos para el presente proyecto han sido cumplidos satisfactoriamente incluyéndose una serie de recomendaciones que deben tenerse en cuenta para el trabajo futuro.

RECOMENDACIONES.

Como hemos observado los objetivos trazados para este trabajo han sido logrados, sin embargo la propuesta es sólo la primera fase de un proyecto que puede ser mucho más ambicioso. Por lo que hacemos las siguientes recomendaciones:

- Ampliar los servicios del portal utilizando las facilidades de los CMS tales como foros de discusión, registrar usuarios que deseen recibir boletines noticiosos o artículos, búsquedas etc.
- Mantener actualizada la versión del Sistema Manejador de Contenido (Xoops) sobre el cual está montado el portal.
- Vinculo al portal de las Misiones Sociales de la Republica Bolivariana por su relación con los temas tratados en Patria Grande.

REFERENCIAS BIBLIOGRÁFICAS.

- [1] Sánchez Germán Otero. Bienvenidos a Patria Grande. Revista Patria Grande No.1 Febrero 2005.
- [2] Sistemas de Gestión de Contenidos: Artículo: Sistemas para administración de contenido de páginas Web: <http://www.webtaller.com/codigo/Clicks.php/29/articulos/>.
- [3] Historia del Web: Artículo: "Breve historia de la World Wide Web". <http://html.conclase.net/articulos/historia>.
- [4] Introducción a los Sistemas de Gestión de Contenidos (CMS) de código abierto (29.11.04) <http://www.uoc.edu/mosaic/articulos/cms1204.html>
- [5] WikiPedia. La enciclopedia Libre: CMS. <http://es.wikipedia.org/wiki/CMS>
- [6] Introducción a los sistemas de gestión de contenidos. <http://geneura.ugr.es/~jmerelo/tutoriales/cms/>
- [7] ¿Robertson, J., So, what is a content management system? [en línea]. Step Two, 3 junio 2003 <http://www.steptwo.com.au/papers/kmc_what/index.html> [Consulta: marzo-abril 2004]
- [8] CMS Sistemas Gestión de Contenidos: Principales Ventajas <http://www.unsitio.com/cms.htm>
- [9] Christian Van Der Henst y Herminio Heredia Santos : Introducción al PHP <http://www.maestrosdelweb.com/editorial/phpintro/>
- [10] Claudio Casares: Curso de SQL. <http://www.programacion.com/tutorial/sql/1/>
- [11] Booch, G., Rumbaugh, J., Jacobson, I. "El Proceso Unificado de Desarrollo de Software ". Addison-Wesley. 2000.
- [12] Tutorial de Desarrollo Orientado a Objetos con UML . <http://www.clikear.com/manuales/uml/>
- [13] Joseph, Schmuller. "Aprendiendo UML en 24 horas", Prentice-Hall, Inc. 2001.
- [14] Productos Racional. Infosevgroup.com. http://www.infosgroup.com/paginas/v4/publico/soluciones/soluciones_producto/rational/productosrational.asp?referal=/paginas/v4/publico/soluciones/soluciones_producto/rational.asp
- [15] De Wikipedia, la enciclopedia libre. MySQL: <http://es.wikipedia.org/wiki/MySQL>
- [16] Miguel Ángel Álvarez. Evaluando Zend Studio. <http://www.maestrosdelweb.com/editorial/zendstudio/>
- [17] Una introducción visual a XOOPS. 2004 The XOOPS Project. <http://www.esxoops.com/docu/newbies/index.html?summaryoffeatures.htm>

[18] Ferran Martí. ¿La era de Internet?. La era de Internet no existe. (20-12-2000)

<http://www.webaprendiz.com/viewp/notas/mkt/2000/12/id0006.htm>

[19] De Wikipedia, la enciclopedia libre: Modelo servidor-cliente.

http://es.wikipedia.org/wiki/Modelo_servidor-cliente

[20] Páginas Estáticas Vs. Dinámicas.

<http://www.desarrolloweb.com/articulos/712.php?manual=27>

BIBLIOGRAFÍA.

1. Sæther Bakken, Stig. Schmid, Egon. Manual de PHP. PHP Documentation Group 2003 <http://www.php.net/docs.php>.
2. Sistemas de Gestión de Contenidos: Artículo: “CMS Quick Guide”. . <http://www.opensourcecms.com/index.php?option=content&task=view&id=388>.
3. Sistemas de Gestión de Contenidos: Tutorial: “CMS Tutorial”. . <http://www.opensourcecms.com/index.php?option=content&task=view&id=500>.
4. Sistemas de Gestión de Contenidos: Artículo: “So what is a CMS?”. . http://typo3.com/What_is_a_CMS.1351.0.html.
5. Sistemas de Gestión de Contenidos: Artículo: “What is a content management system?”. (06/2003) http://www.steptwo.com.au/papers/kmc_what/.
6. Sistemas de Gestión de Contenidos: Artículo: Sistemas para administración de contenido de páginas Web: <http://www.webtaller.com/codigo/Clicks.php/29/articulos/>.
7. Sistemas de Gestión de Contenidos: “Roadmap”. <http://www.esxoops.com/modules/news/>.
8. Sistemas de Gestión de Contenidos: “Operation guide”. Xoops Documentation Team. <http://docs.xoops.org/modules/xdocman/index.php?doc=xd-01&lang=en>
9. Sistemas de Gestión de Contenidos: “API Doc”. <http://www.xoops.org/misc/api/>
10. Sistemas de Gestión de Contenidos: “Drupal”. CmsMatrix.org (2005). <http://www.cmsmatrix.org/matrix?func=viewDetail&wid=2&listingId=1050>.
11. Sistemas de Gestión de Contenidos: “XOOPS”. CmsMatrix.org (2005). <http://www.cmsmatrix.org/matrix?func=viewDetail&wid=2&listingId=1051>.

12. Sistemas de Gestión de Contenidos: “Apache Lenya”. CmsMatrix.org (2005). <http://www.cmsmatrix.org/matrix?func=viewDetail&wid=2&listingId=1123>.
13. Sistemas de Gestión de Contenidos: “Mambo”. CmsMatrix.org (2005). <http://www.cmsmatrix.org/matrix?func=viewDetail&wid=2&listingId=1074>.
14. Sistemas de Gestión de Contenidos: “TYPO3”. CmsMatrix.org (2005). <http://www.cmsmatrix.org/matrix?func=viewDetail&wid=2&listingId=1023>.
15. Sistemas de Gestión de Contenidos: “Feature List” (2005). CmsReview.com. <http://www.cmsreview.com/Features/Lists.html>.
16. Sistemas de Gestión de Contenidos: “Apache Lenya”. Apache Software Foundation (2005). <http://lenya.apache.org/>.
17. Sistemas de Gestión de Contenidos: “LifeRay Portal”. LifeRay Enterprise (2005). <http://www.liferay.com/cms/servlet/PRODUCTS-PORTAL>
18. Pressman, R. “*Software Engineering. A Practitioner’s Approach*”. Fourth Edition. McGraw – Hill. USA, 1999.
19. Booch, G., Rumbaugh, J., Jacobson, I. “*El Lenguaje Unificado de Modelado*”. Addison-Wesley. 1999.
20. Larman, C. “*Applying UML and Patterns. An Introduction to Object-Oriented Analysis and Design*”. Prentice-Hall, Inc. 1998.
21. Lenguajes de Programación: “Programación Web”. <http://lenguajes-de-programacion.com/programacion-web.shtml>.
22. Desarrollo de Web: Manual: “*Qué es cada tecnología*”. (02-2005). <http://www.desarrolloweb.com/manuales/15/>.
23. Desarrollo de Web: Manual: “*Tutorial de SQL*”. (02-2005). <http://www.desarrolloweb.com/manuales/9/>.

24. Desarrollo de Web: Artículo: “*Páginas dinámicas*”. (02-2005).
[http://www.desarrolloweb.com/manuales/7/.](http://www.desarrolloweb.com/manuales/7/)
25. Introducción a los sistemas de gestión de contenidos.
<http://geneura.ugr.es/~jmerelo/tutoriales/cms/>
26. Robertson, J., How to evaluate a content management systempers/kmc_evaluate/index.html. [en línea]. Step Two, 23 Enero 2002
http://www.steptwo.com.au/papers/kmc_evaluate/index.html

GLOSARIO DE TÉRMINOS.

Acónimo: diccionario de sinónimos; lista de sinónimos que una máquina de búsqueda puede emplear para determinar coincidencias con determinadas palabras, si las palabras mismas no se encuentran en el documento.

CSS: (Hoja de Estilo en Cascada) Dentro del diseño de páginas de Internet se presenta esta como la vanguardia en cuanto a definición de estilos dentro de las plantillas de diseño. A través de instrucciones en código HTML se definen los estándares del conjunto de páginas que conforman el proyecto. La meta es uniformizar el diseño.

CGI: (Interfaz Común de Pasarela) Es un protocolo o interfaz de intercambio de información que se realiza entre el navegador del usuario y un servidor WWW.

Código Abierto: Es una tendencia internacional del desarrollo de software que profesa la distribución del código junto a las aplicaciones, se rigen por licencias tales como GNU/GPL.

HTTP: Es el conjunto de reglas para intercambiar archivos (texto, gráfica, imágenes, sonido, video y otros archivos multimedia) en la World Wide Web.

IDE: Ambiente de desarrollo integrado. Es como se le llama al ambiente que proporciona al usuario una determinada herramienta de desarrollo.

ISAPI: Interfaz de programación de aplicaciones de servidor para Internet. Una interfaz para el desarrollo de aplicaciones de servidor Web, desarrollado por Process Software y Microsoft Corporation, que se utiliza en lugar de CGI.

Microsoft: Compañía de software más grande del mundo. Fue fundada en 1975 por Paul Allen y Bill Gates. Aunque también se conoce por sus lenguajes de

programación y aplicaciones para computadores personales, el éxito sobresaliente de Microsoft se debe a sus sistemas operativos DOS y Windows.

OMT: (Object Modeling Techniques). Metodología de análisis y diseño orientado a objetos, desarrollada por Jim Rumbaugh; es una de las tres grandes metodologías que contribuyeron al desarrollo del RUP.

OOSE: (Object Oriented Software Engineering). Metodología de análisis y diseño orientado a objetos, desarrollada por Ivar Jacobson; es uno de los precursores del UML.

Unix: Sistema operativo atribuido a Ken Thompson y comercializado por la empresa ATT en la década de los 70s que alcanzó mucho éxito, sobretodo en las universidades y posteriormente en las empresas. Entre sus principales características tenemos que es: portable, robusto, flexible y abierto, actualmente goza de gran popularidad dentro de la tecnología de Internet.

Web Services: aplicación simple que realiza un cometido y que puede formar parte de otros servicios para formar un servicio más completo.

Zend: Compañía líder de infraestructuras para web; está reconocida internacionalmente como la autoridad actualmente en PHP. Sus fundadores son los diseñadores del PHP v.4 en adelante, actualmente es una compañía líder dentro de la comunidad Open Source.

ANEXOS.

Anexo 1

Término	Explicación
Autorías (Audit Trail)	Mecanismo de auditorías, que permite grabar lo eventos que ocurren en el sistema.
Aprobación de Contenidos (Content Approval)	Mecanismo global al sistema para aprobar nuevos contenidos o cambios en general (incluye diseño)
Granularidad de los permisos (Granular Privileges)	Cómo se gana autorización para leer, modificar, un elemento del sistema (página, componente, etc.)
Autenticación (Authentication)	Módulos de autenticación del sistema. Debe verificar: LDAP, Kerberos, NIS, NTLM, y también sistemas ad-hoc (en una DB, por ejemplo)
Notificaciones (Notifications)	Mecanismo para notificar sucesos: contenidos pendientes, problemas detectados, etc...
Arena (Sandbox)	Mecanismo que aísla los cambios del sitio online (relación con Content Approval y Staging)
SSL	¿Se puede configurar el sistema, parte del sistema, o páginas públicas por SSL?
Control de versiones (RCS)	Soporte de versiones de los componentes del sistema (incluir componentes, layouts, etc...)
Drag-N-Drop	Posibilidad de usar el Drag-N-Drop en la edición del contenido
URL legibles (Friendly URLs)	URL legibles para el usuario
Corrector ortográfico (Spell Checker)	Tiene corrector ortográfico para la edición de contenido.
Plantillas (Template Language/Support)	Existe un mecanismo de plantillas para diseño y contenidos. Incluir si es integrable a IDE y manejable (como elemento) dentro del sistema.
Cronograma (Content Scheduling)	Los contenidos/diseños del sitio se pueden programar por horarios
Sub Webs / Sub Sites	Permite la inclusión de sub-sitios auto manejados
Estadísticas (Web Stats)	Reportes de estadísticas (ejemplo página más visitada, etc.).
Reutilización de contenidos (Content Reuse)	Permite la incorporación del mismo elemento en varias páginas

Static Export	Capacidad de exportar el contenido como HTML estático
Asset Management	Almacenamiento central de ficheros, imágenes, etc.
Representación de los contenidos (Content Staging)	Las modificaciones se realizan en un servidor (staging server) y luego son fácilmente actualizadas en el servidor online.
Administración integrada (Inline Admin)	El contenido se edita directamente en la página donde se coloca.
Gestor de flujo de trabajo (Workflow Engine)	Herramienta para controlar el proceso de creación de los contenidos, antes del Content Approval.
Metadatos (Metadata)	Capacidad de asociar metadatos a los contenidos.
Wiki Aware	Sistema que adopta conceptos wiki.
WebServices Frontend	El sistema es accesible mediante servicios Web.
Multi-Site Deployment	Habilidad de mantener varios sitios con el mismo CMS (Relación con Content Reuse)

Tabla 1 Criterios de Evaluación

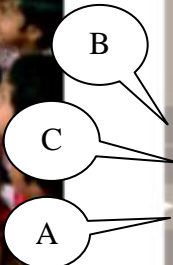
Anexo 2

CU-10	Mostrar sección de imágenes.
Actores	Usuario Anónimo, Administrador, Editor
Propósito	Que el usuario pueda ver las imágenes publicadas en el portal.
Resumen	
<p>Este caso de uso inicia cuando un usuario accede al portal como usuario anónimo en la Internet o una intranet, entonces este se encarga de mostrar la galería de imágenes (<i>Pantalla 1-A</i>), la criatura diaria del sitio (<i>Pantalla 1-B</i>) y la imagen para coleccionar como fondo de pantalla (<i>Pantalla 1-C</i>).</p>	
Referencias	R11
Precondiciones	
Poscondiciones	
Requisitos especiales	

Prototipo



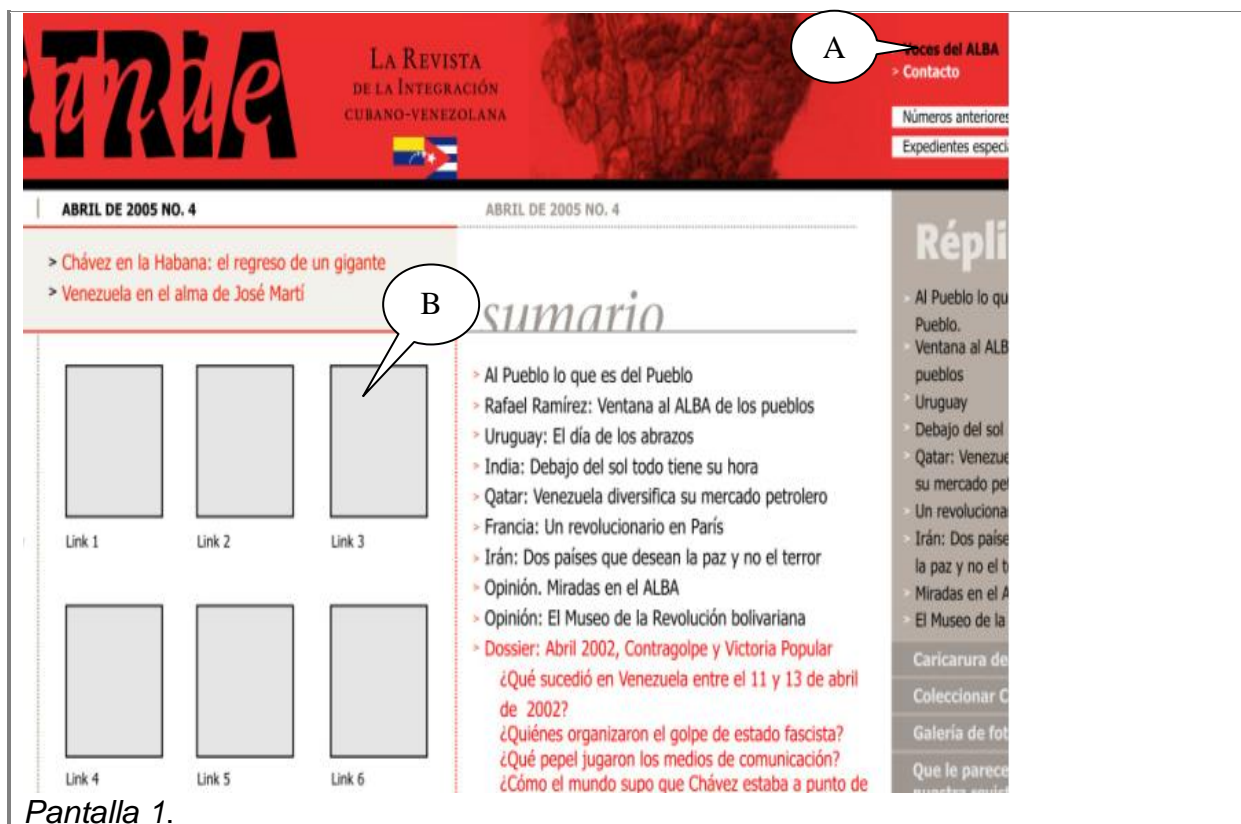
- Ventana al ALBA de los pueblos
- Uruguay
- Debajo del sol
- Qatar: Venezuela diversifica su mercado petrolero
- Un revolucionario en París
- Irán: Dos países que desean la paz y no el terror
- Miradas en el ALBA
- El Museo de la Revolución
- Caricatura del día
- Coleccionar Cartel
- Galería de fotos
- Que le parece a usted nuestra revista:



Pantalla 1.

Anexo 3

CU-9	Mostrar enlaces a otros sitios
Actores	Usuario Anónimo, Administrador, Editor
Propósito	Que el usuario pueda ver u usar enlaces a otros sitios de interés para la revista.
Resumen	Este caso de uso inicia cuando un usuario accede al portal como usuario anónimo en la Internet o una intranet, entonces este se encarga de mostrar mediante un enlace (Voces del Alba) (Pantalla 1-A) los sitios a los que se puede acceder desde el portal de Patria Grande (Pantalla 1-B).
Referencias	R12
Precondiciones	¿?
Poscondiciones	¿?
Requisitos especiales	¿?
Prototipo	



Pantalla 1.

Anexo 4

CU-11	Mostrar encuestas.
Actores	Usuario Anónimo, Administrador, Editor
Propósito	Que el usuario pueda ver y participar de las encuestas publicadas en el portal.
Resumen	Este caso de uso inicia cuando un usuario accede al portal como usuario anónimo en la Internet o una intranet, entonces este se encarga de mostrar la encuesta publicada (Pantalla 1-A) así como ver los resultados de la misma (Pantalla 1-B).
Referencias	R13
Precondiciones	
Poscondiciones	
Requisitos especiales	

Prototipo

Venezuela diversifica su mercado petrolero
Un revolucionario en París
Dos países que desean la paz y no el terror
Miradas en el ALBA
El Museo de la Revolución bolivariana
Abril 2002, Contragolpe y Victoria Popular
¿Qué sucedió en Venezuela entre el 11 y 13 de abril de 2002?
¿Quiénes organizaron el golpe de estado fascista?
¿Qué papel jugaron los medios de comunicación?
¿Cómo el mundo supo que Chávez estaba asesinado por los golpistas?
El nuevo imperio romano frente al terrorismo
¿Qué decía Ernesto
Crisis: Venezuela sin Chávez es como la teoría de la

su mercado petrolero

- > Un revolucionario en París
- > Irán: Dos países que desean la paz y no el terror
- > Miradas en el ALBA
- > El Museo de la Revolución

Caricatura del día

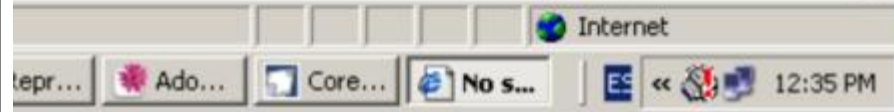
Coleccionar Cartel

Galería de fotos

Que le parece a usted nuestra revista:


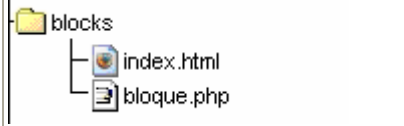



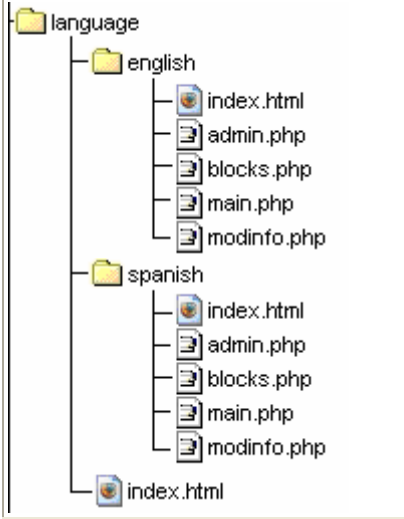
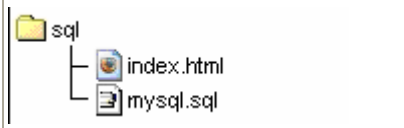
- Muy buena
- Interesante
- Mala

VOTAR VER RESULTADOS



Pantalla 1.

Anexo 5

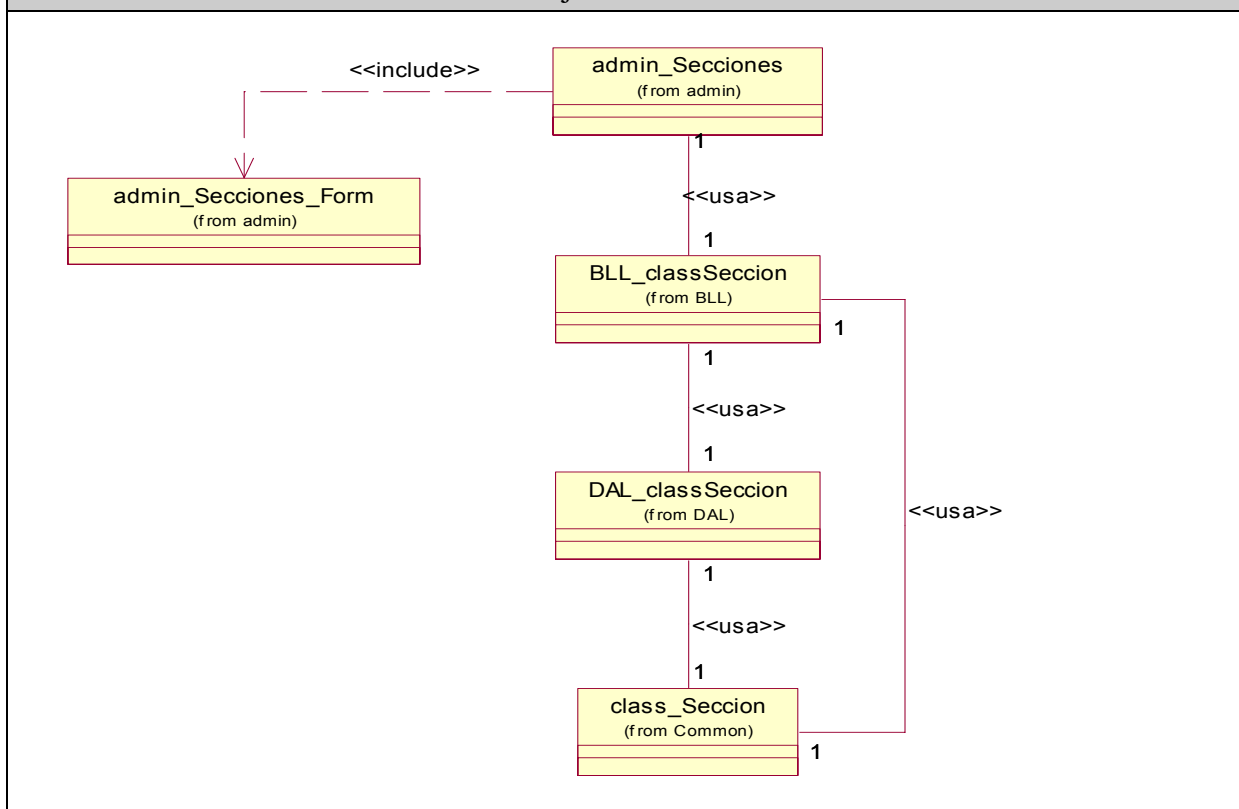
	<p>admin: contiene los archivos que sólo podrá ver quién posea permisos de administración sobre el módulo.</p> <p>-index.php: puede contener un menú o directamente la administración del módulo.</p> <p>-menu.php: contiene los links que aparecerán en el icono de administración, al pasar el mouse sobre él.</p> <p>-administracion.php: puede ser el archivo de administración. Su nombre puede ser distinto, sólo debe ser indicado en <i>menu.php</i> con el nombre indicado.</p>
	<p>blocks: contine los archivos que poseen los posibles bloques que tenga el sistema.</p> <p>-bloque.php: podrá tener el contenido del bloque que posea el módulo.</p>
	<p>class: si estas programando con orientación a objetos, es conveniente que guardes dicho código en diferentes archivos contenidos en esta carpeta. Luego para usar las clases creadas sólo deberás incluir el archivo.</p>
	<p>images: es la carpeta donde podrás almacenar tus imágenes.</p> <p>Generalmente, slogopng es el icono de administración.</p>
	<p>include: en esta carpeta podrás guardar todos aquellos archivos que contienen funciones que te permitirán la implementación del módulo. Luego para usarlo sólo deberás incluirlo en donde desees.</p> <p>-search.inc.php: corresponde al archivo que contiene la función de búsqueda del módulo y que será utilizada por el buscador global que posee XOOPS.</p>
	<p>language: esta carpeta contendrá las carpetas de lenguajes que crees para el módulo. Es conveniente siempre tener creada la carpeta english con sus archivos aunque no estén bien traducidos, pues es el lenguaje original de XOOPS y tal vez alguien más en el futuro los pueda traducir.</p> <p>-admin.php: contiene el lenguaje que será incluido para la administración. Si tu módulo no posee administración no es necesario este archivo.</p> <p>-blocks: contiene el lenguaje para los bloques si los haz creado. En caso contrario no es necesario tener este archivo.</p> <p>-main.php: contiene el lenguaje que será incluido en la presentación al usuario del módulo. Si tu módulo sólo posee administración, puedes omitir este archivo.</p> <p>-modinfo.php: contiene el lenguaje que será utilizado principalmente en la descripción del sistema. Sus constantes serán incluidas en la creación del archivo <i>xoops_version.php</i>.</p>
	<p>sql: esta carpeta contiene los scripts sql que posea el módulo. Así se incluye en esta carpeta el archivo <i>mysql.sql</i>, que corresponde a los scripts de creación de tablas e inserciones básicas para el servidor de bases de datos MySQL, que hasta el momento es sobre el que se soporta nuestro XOOPS.</p>

	<p>templates: los templates son plantillas, HTML principalmente, en las cuales se presentarán los datos que el módulo entregue al usuario. El módulo puede tener o no templates, todo depende de cómo el programador desee crear su módulo. En caso de poseer templates, es conveniente poner nombres descriptivos, por ejemplo, la página <i>index.php</i> posee el template <i>theme_index.html</i></p>
	<p>Archivos bases del Módulo: aquí se encuentran los archivos que permitirán instalar el módulo y enlazarlo con XOOPS. Así, en <i>header.php</i> podremos incluir todos aquellos archivos que requiera el módulo.</p> <p>-index.php: este archivo entregará al usuario el acceso a los datos que entregue el módulo. Si el módulo sólo es para administración, puede omitir este archivo, caso en el cual deberá incluir un archivo index.html que explicaremos más adelante.</p> <p>-xoops_version.php: es el archivo que contiene toda la información referente al módulo. Siempre debe existir, pues sin él el módulo no podrá ser instalado.</p>

Anexo 6

Sub-Paquete 2.4 Manejar Contenido de Secciones.

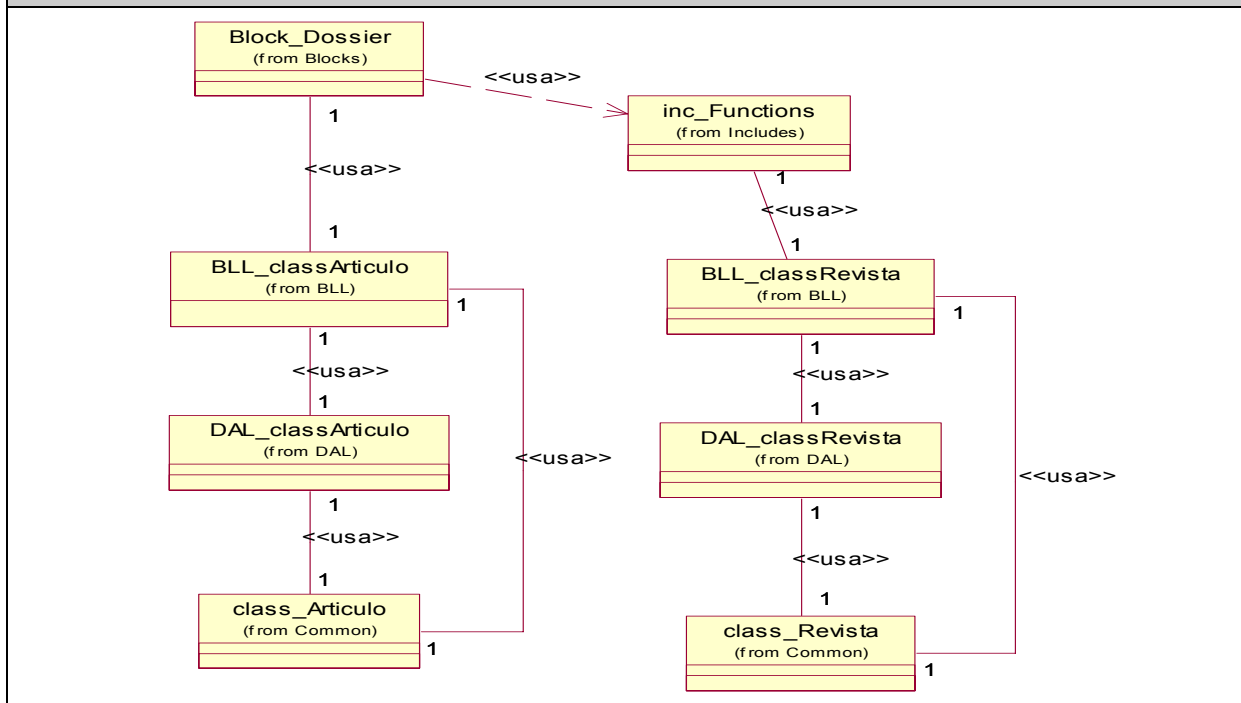
DIAGRAMA DE CLASES WEB: Manejar Contenido Secciones.



Anexo 7

Sub-Paquete 3.3 Mostrar Colección Dossier.

DIAGRAMA DE CLASES WEB: Mostrar Colección Dossier.



Anexo 8

Sub-Paquete 3.4 Mostrar Ediciones Anteriores.

DIAGRAMA DE CLASES WEB: Visualizar Ediciones Anteriores.

