

Universidad de las Ciencias Informáticas

Facultad 1



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

Título: “Portal Web de la Revista Pionero”

Autores:

Osdany Toledo Reyes

Agustin Medina Torrez

Tutores:

Ing. Yuneldis Reyes Velázquez

Ing. Ariagna González Landeiro

DECLARACIÓN DE AUTORÍA

Declaramos ser autores de la presente tesis y reconocemos a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo. Para que así conste firmamos la presente a los ____ días del mes de _____ del año 2013.

Firma del Autor
Agustin Medina Torrez

Firma del Autor
Osdany Toledo Reyes

Firma del Tutor
Ing. Yuneldis Reyes Velázquez

Firma del Tutor
Ing. Ariagna González Landeiro



“Nadie ha hecho tanto en tan poco tiempo.”

Fidel Castro Ruz

AGRADECIMIENTOS

Le agradezco mi tesis a todo el mundo, en especial a mí.

NOMBRE DEL TESISISTA

Osdany Toledo Reyes.

DEDICATORIA

Les dedico mi tesis a mi mamá por ayudarme a seguir adelante en mi carrera todos estos años y a mi papá por confiar en mí y saber que yo sí podía graduarme.

NOMBRE DEL TESISISTA

Osdany Toledo Reyes.

Resumen

Resumen

La revista digital Pionero, es una solución informática sobre tecnologías Web, mediante la utilización del Gestor de Contenido (CMS) Drupal, por las potencialidades que brinda para el manejo y gestión del contenido, permite además la correcta informatización del proceso de edición de los diferentes elementos relacionados con la revista, adaptándose a las características de diferentes medios de divulgación, por ser un sistema modular y flexible. El sistema desarrollado posibilita controlar y agilizar la publicación de materiales dedicados a los jóvenes lo que comprende la creación, edición, aprobación y publicación de contenidos. Se pretende convertir el sistema en un entorno de trabajo sencillo y eficiente, donde los involucrados desarrollen la actividad fundamental del proceso editorial y los lectores puedan consultar la información que les resulte útil. Los artefactos y actividades basados en la metodología utilizada OpenUP permiten una mejor organización del proceso de desarrollo de software del portal de la revista Pionero. Las pruebas que se le realizan a la aplicación permiten obtener un producto con el menor número de errores posibles, satisfaciendo así las necesidades de los usuarios.

Palabras Clave: artefactos, Drupal, modular, revista.

Índice

Índice

Capítulo 1. Fundamentación teórica.....	6
1.1 ¿Qué es un portal Web?.....	6
1.2 Tendencias de revistas y portales Web.....	6
1.2.1 Los portales Web en el mundo	6
1.2.2 Los portales Web en Cuba	7
1.2.3 Tendencias actuales en la publicación de portales Web	9
1.3 Sistema de Gestión de Contenidos (CMS).....	9
1.3.1 CMS Joomla	10
1.3.2 CMS Drupal	11
1.4 Lenguajes empleados.....	13
1.4.1 Lenguaje Unificado de Modelado 2.0 (UML)	13
1.4.2 Lenguaje de Marcado de Hipertexto 4.0 (HTML).....	13
1.4.3 Hojas de Estilos en Cascada (CSS)	13
1.4.4 Lenguaje de Programación PHP 5.0	14
1.4.5 Lenguaje de Programación <i>JavaScript</i>	15
1.5 Entorno de Desarrollo Integrado (IDE)	15
1.5.1 NetBeans 7.1	15
1.5.2 <i>Apache</i> Web Server 2.2.8.....	16

Índice

1.6	Sistemas Gestores de Base de Datos	16
1.6.1	MySQL	17
1.6.2	PostgreSQL 8.4	17
1.7	Metodología a utilizar.....	18
1.7.1	eXtremeProgramming (XP).....	18
1.7.2	Scrum.....	19
1.7.3	Metodología de Desarrollo de Software OpenUP	19
1.8	Herramientas Utilizadas.....	20
1.8.1	Pgadmin III	20
1.8.2	Visual Paradigm para UML 8.0	20
Capítulo 2. Características del Sistema		22
2.1	Modelo de Dominio.....	22
2.1.1	Diagrama de Clases del Modelo del Dominio	22
2.2	Especificación de los requisitos del sistema.....	23
2.2.1	Requisitos Funcionales	24
2.2.2	Requisitos No Funcionales	30
2.3	Descripción del sistema propuesto.....	32
2.4	Descripción de Requisitos de Software	32
Capítulo 3. Análisis y Diseño		41

Índice

3.1	Patrones de Arquitectura y Diseño	41
3.1.1	Patrones de arquitectura.....	41
3.1.2	Patrones de Diseño	43
3.2	Diagramas de Clases de Diseño	45
3.3	Diagramas de Secuencia.....	47
3.4	Diagrama de Despliegue	48
Capítulo 4. Implementación y Pruebas.....		50
4.1	Diagrama de componentes.....	50
4.2	Diseño de la Base de Datos	50
4.2.1	Modelo de Datos	51
4.3	Código fuente	51
4.4	Pantallas principales de la aplicación	53
4.5	Validación del sistema	56
4.5.1	Pruebas Funcionales	56
4.5.2	Pruebas de carga y estrés de la aplicación	59
4.5.3	Pruebas de Seguridad	60
Conclusiones		62
Recomendaciones		63
Referencias Bibliográficas		64

Índice

Anexos.....	67
5.1 Anexos 1.....	67
5.2 Anexos 2.....	76

Introducción

El surgimiento de la informática ha irrumpido vertiginosamente en el desarrollo de la sociedad actual y ha beneficiado a la mayor parte de las ramas de la actividad social, económica y política del mundo; propiciando con ello la informatización de la sociedad entendido como “el proceso de utilización ordenada y masiva de las Tecnologías de la Información y las Comunicaciones en la vida cotidiana, para satisfacer las necesidades de todas las esferas de la sociedad, en su esfuerzo por lograr cada vez más eficacia y eficiencia en los procesos y por consiguiente mayor generación de riqueza y aumento en la calidad de vida de los ciudadanos”. [1]

Un salto trascendental en el desarrollo de las TIC, lo constituyó el surgimiento de Internet, como sistema mundial de redes de computadoras, integrado por las diferentes redes de cada país donde un usuario, desde una computadora, puede acceder a la información de otra y tener comunicación directa con personas en cualquier lugar del mundo. Marcando el inicio de una nueva era en el desarrollo de las comunicaciones.

A diario son millones los usuarios que acceden a la red de redes en busca de información, usando de forma variada las potencialidades que brinda Internet. En este sentido los portales Web se han convertido en una fortaleza, evolucionando con el desarrollo de Internet, pues han pasado de ser desarrollados de forma estática, enfocados principalmente en mostrar información permanente y contruidos básicamente solo por hipervínculos o enlaces (*links*) a ser desarrollados dinámicamente, mostrando las potencialidades de la Web 2.0, propiciando servicios informativos e interactivos de forma eficiente como pueden ser foros, noticias, encuestas, buscadores, transacciones electrónicas o de manera general información en distintos formatos. La gestión de la información se ha convertido en una necesidad para las empresas en desarrollo, posibilitando así que la sociedad esté actualizando información de lo que sucede en diferentes lugares del mundo, satisfaciendo las necesidades de los usuarios y los intereses de las organizaciones. Hoy en día los portales Web pueden llegar a ser muy complejos, gestionando gran cantidad de información y brindando variadas funcionalidades al usuario final. [2]

Cuba en el esfuerzo de informatizar la sociedad, durante varios años se ha enfocado en el desarrollo de portales Web para divulgar información acerca de la realidad cubana, así como mantener informada a la población de lo que está ocurriendo a nivel nacional e internacional. Ejemplo de ellas se encuentra el sitio Web de la revista Pionero, la misma fue fundada el 25 de noviembre de 1961 en formato duro donde su directora recuerda su aparición como tabloide con una frecuencia semanal, destinado hasta 1990 a los

Introducción

escolares de la enseñanza primaria y secundaria, para reflejar en un lenguaje asequible el acontecer de la vida de los pioneros (persona que realiza los primeros descubrimientos o los primeros trabajos en una actividad determinada) en todo el país. [3]

Con el objetivo de llegar a un mayor número de clientes, teniéndose en cuenta la cifra finita de copias físicas que se podían distribuir cada vez que salía un número de la revista, así como la necesidad de darle a conocer al mundo la realidad pioneril, se decide crear en el año 2000 la versión digital de la misma. Este nuevo formato permitía insertar un mayor número de publicaciones que el tabloide en formato duro, lo que brindaba mayores posibilidades al usuario final. Teniéndose en cuenta las características de la Web en el momento que surge la versión digital, se puede decir que aunque cumplió sus objetivos fue desarrollada en una Web estática, con pocas posibilidades de colaboración, lo que hace que hoy no esté acorde ni asimile los nuevos métodos comunicacionales que impone la Web 2.0. [4]

Actualmente el sitio Web de la revista pionero no permite el registro de usuarios, lo que trae consigo que los mismos no puedan realizar comentarios ni dar opiniones acerca de contenidos publicados en la revista. No se muestran trazas para ubicar a los usuarios dentro del sitio y existen vínculos rotos lo que conlleva a la pérdida de la navegación y por consiguiente que no se logre divulgar correctamente la información de la revista. La inexistencia de un buscador trae problemas para encontrar de manera rápida y ágil los contenidos publicados. Existen contenidos con formatos de fuentes distintos lo cual le quita presencia y prestigio haciendo que no sea amigable a la hora del usuario interactuar con el mismo. Cuenta con problemas estructurales en la organización de la información y la construcción, desmotivando a los usuarios del acceso al sitio en busca de la información que necesitan.

La siguiente investigación tiene como **Problema Científico**: ¿Cómo contribuir a la divulgación y promoción del contenido de la revista Pionero?

Constituyendo el **Objeto de Estudio** los procesos de gestión de los contenidos en los portales Web actuando en el **Campo de Acción** los portales Web dedicados a la publicación de revistas digitales.

En el presente Trabajo de Diploma se pretende dar solución al problema anteriormente expuesto, presentando como **Objetivo General** desarrollar un portal Web para la gestión de los contenidos de la revista Pionero teniendo en cuenta los beneficios de la Web 2.0.

Introducción

Se establecen como **Objetivos Específicos** de la investigación:

- Caracterizar las tendencias actuales en el desarrollo de los portales Web dedicados a la difusión de revistas.
- Diseñar el portal Web de la revista pionero teniéndose en cuenta las funcionalidades con las que debe contar.
- Implementar las funcionalidades y servicios definidos para el portal Web de la revista Pionero.
- Validar las funcionalidades del portal Web de la revista Pionero.

En la materialización de este trabajo se tiene como **Idea a Defender** el desarrollo de un portal Web para la gestión de los contenidos de la revista Pionero teniéndose en cuenta que los beneficios de la Web 2.0 contribuirá a la divulgación y promoción del contenido de la revista Pionero.

Para dar cumplimiento a los objetivos trazados, se diseñaron las siguientes **Tareas**:

- Caracterización de las tendencias actuales en el desarrollo de los portales Web para la difusión de revistas digitales.
- Selección de las tecnologías, herramientas, estándares, patrones y metodologías que permitan desarrollar el portal Web.
- Identificación de los requisitos funcionales y no funcionales a incluir en el portal Web de la revista Pionero.
- Diseño de la arquitectura de la información del portal Web para la revista Pionero.
- Validación de los requisitos y los acuerdos tomados entre clientes y grupo de proyecto aplicando las técnicas de validación definidas.
- Gestión de los requisitos y elaboración del código funcional a partir de la codificación y el diseño de algoritmos garantizando el cumplimiento de los requerimientos del problema.
- Implementación de las funcionalidades del portal Web para la revista.
- Aplicación de las pruebas de funcionalidad al portal Web de la revista Pionero.

Introducción

- Aplicación de las pruebas de carga y estrés al portal Web de la revista Pionero.
- Aplicación de las pruebas de seguridad al portal Web de la revista Pionero.

Para la ejecución de la presente investigación se utilizaron los siguientes **Métodos de Investigación:**

Métodos Teóricos:

Analítico-sintético: El mismo se enfocó en determinar la esencia de los fenómenos, los rasgos que los caracterizan y los distinguen, siendo utilizado para determinar las características comunes y específicas de las herramientas de gestión de conocimiento y para hacer un análisis detallado de todos los procesos para la gestión del conocimiento en línea y a partir de este análisis diseñar un sistema que cumpla con las necesidades planteadas.

Análisis Histórico Lógico: En la primera parte de la investigación se desarrolló un estudio de la trayectoria histórica, la evolución y el desarrollo de revistas y sus portales homólogos desde su surgimiento hasta la actualidad, se analizaron las ventajas y desventajas de cada una de las herramientas para su posterior utilización en el desarrollo de la aplicación.

Modelación: Se utilizó para formular un listado del contenido con el que debería contar la aplicación, estructurar el sistema y modelar las relaciones entre entidades en los diferentes diagramas.

Métodos Empíricos:

Entrevistas: Como parte del desarrollo del portal Web se llevó a cabo diferentes entrevistas con los clientes permitiendo el levantamiento de los requerimientos tanto funcionales, como no funcionales para la implementación del portal Web.

El presente documento se encuentra estructurado en 4 capítulos:

Introducción

Capítulo 1: Fundamentación teórica

El capítulo abordará las tendencias actuales en cuanto a la gestión de contenidos de portales Web de revistas a nivel nacional e internacional, así como el análisis de las principales herramientas, lenguajes y metodología a utilizar en el desarrollo de la aplicación.

Capítulo 2: Características del sistema

En el capítulo se definen las características que posee el sistema que será implementado como solución. Se realiza además el modelo de dominio donde se definen los diferentes conceptos que forman el entorno donde se desarrolla la aplicación así como la descripción de sus clases. Se analizan inicialmente las entidades involucradas y se listan los requisitos que deben desarrollarse.

Capítulo 3: Análisis y diseño

En el capítulo se concretarán los requisitos funcionales y no funcionales que debe cumplir el sistema, así como las reglas que se deberán tener en cuenta para diseñar la aplicación, representándose mediante diferentes artefactos como: diagramas de clases del diseño, diagramas de secuencia y diagrama de despliegue.

Capítulo 4: Implementación y pruebas

En el capítulo se realiza el desarrollo de la aplicación teniendo en cuenta el modelado que se obtuvo del análisis y diseño, dando cumplimiento a todos los requisitos trazados y se efectúan las pruebas requeridas para la comprobación del sistema automatizado y las funcionalidades de la revista.

Capítulo 1. Fundamentación teórica

Capítulo 1. Fundamentación teórica

El siguiente capítulo abordará el estudio de sistemas homólogos, para incluir algunas características necesarias para la correcta implementación del portal Web de la revista Pionero. También se analizarán las principales metodologías, herramientas y lenguajes a utilizar en el desarrollo de la solución.

1.1 ¿Qué es un portal Web?

Portal es un término, sinónimo de puente, para referirse a un sitio Web que sirve o pretende servir como un sitio principal de partida para las personas que se conectan al *World Wide Web*. Los portales tienen gran reconocimiento en Internet por el poder de influencia que poseen sobre grandes comunidades. Son empleados para localizar la información y de ahí comenzar alguna actividad en Internet. Un portal es más bien una plataforma de despegue para la navegación en la Web que ofrece al usuario, de forma fácil e integrada, el acceso a una serie de recursos y servicios relacionados a un mismo tema. [5]

1.2 Tendencias de revistas y portales Web

Actualmente existen una gran cantidad de revistas y portales Web pertenecientes a instituciones y asociaciones, los cuales se encuentran distribuidos por todo el mundo satisfaciendo las necesidades de los usuarios y los intereses de sus empresas; a continuación se presentan algunos ejemplos, de los que se realizará un estudio de sus tendencias para incorporar características necesarias para un correcto desarrollo del portal Web de la revista Pionero.

1.2.1 Los portales Web en el mundo

Portal de la Asociación Internacional para el Reconocimiento de Patrones (IAPR)

La Asociación Internacional para el Reconocimiento de Patrones (IAPR) es una asociación internacional de organizaciones con alcance nacional e internacional y se ocupa del reconocimiento de patrones. Normalmente se admite una organización de un mismo país, y las personas interesadas en participar en las actividades de la IAPR pueden hacerlo uniéndose a su organización nacional. Dicha asociación tiene un portal con gran cantidad de visitas a nivel mundial. El mismo se caracteriza por un diseño sencillo, su

Capítulo 1. Fundamentación teórica

facilidad de uso y la correcta estructuración de sus contenidos. Presenta algunos servicios de gran importancia como son la publicación de eventos y de boletines científicos.

Revista El Templo de las Mil Puertas

Revista *online* gratuita de literatura juvenil. Abarca todos los temas relacionados con la literatura juvenil y que pueden interesar a sus lectores: reseñas, entrevistas y reportajes. Cuenta además con un catálogo de novedades, donde pueden descargarse en PDF los números existentes desde el año 2008, la revista Web incluye secciones muy variopintas, desde artículos y premios literarios, hasta recomendaciones, e incluso una sección donde admite colaboraciones de particulares, siempre que sean textos perfectamente redactados.

Revista Dibús

La revista de los jóvenes artistas. Su objetivo es despertar al artista que los lectores llevan dentro a través de las secciones de dibujo, cómics, concursos y reportajes.

1.2.2 Los portales Web en Cuba

Portal CubaSi

Portal que se destaca por la actualización de su información en formato de noticias y multimedias acerca de Cuba y el mundo. El mismo fue diseñado y desarrollado sobre el CMS Joomla por un equipo de desarrolladores de la Empresa de Telecomunicaciones de Cuba (ETECSA) y está basado 10 % en software libre, con licencia GNU. Además ofrece variados servicios tales como la transmisión de emisoras de radio y canales de televisión, búsqueda de contenidos, espacios para anuncios publicitarios de negocios, entre otros.

Portal de la revista digital Cubahora

Cubahora es la primera revista cubana exclusivamente digital que se inserta en el escenario mediático de Internet desde 1998 con el propósito de ampliar la presencia del país en espacios comunicativos virtuales, a partir del tratamiento de temas de actualidad nacional e internacional. Pretende analizar, explicar y argumentar sucesos, sus causas y contextos, así como sus repercusiones; de manera tal que ofrezcan a la

Capítulo 1. Fundamentación teórica

opinión pública la posibilidad no sólo de estar informada sino de conocer y crear juicios de valor sobre cultura, economía, deporte, política, ciencia y técnica, así como coberturas especiales, desde sus propios referentes socioculturales y políticos.

Revista Pionero

Revista que mantiene a los niños, adolescentes y pioneros cubanos al tanto de lo que acontece a nivel nacional e internacional, cuyo principal objetivo es dar a conocer la realidad pioneril cubana, para evitar que el resto del mundo conozca una información tergiversada o completamente distante de la realidad. Cuenta con características como: Publicación de noticias; galerías de imágenes, también cuenta con diferentes secciones: sexualidad, humor y concursos; y cuenta además con la descarga de archivos. Pionero ha contribuido a inculcar en las más jóvenes generaciones los valores del socialismo entre los que destacan el amor a la patria y la disposición a defenderla hasta con la propia vida si fuese necesario. Ha mostrado la participación de nuestra niñez en las tareas de la Revolución y el desempeño de la Organización de Pioneros en sus diferentes etapas; incluidas las relaciones con las organizaciones de pioneros de los desaparecidos países socialistas. Dentro de las tareas de la Organización de Pioneros ha reflejado el trabajo del Movimiento de Pioneros Exploradores, y ha contribuido a orientar a pioneros y guías al respecto, conociendo que esta es una vía de preparar a nuestra niñez para la defensa del país y para la vida en condiciones de campaña. A su vez, ha dedicado atención al movimiento de círculos de interés desde su creación y entre ellos ha mostrado los que tienen un carácter patriótico militar. En vista a celebrarse el cincuenta aniversario de la fundación de la revista, el 25 de noviembre de 2011, el Museo Postal de Cuba le rindió homenaje con la cancelación de un sello conmemorativo. En la estampilla se reflejaron personajes representativos de las historietas que aparecen cada mes en la publicación. Esta publicación recibió en 1968 mención especial por su cooperación en la divulgación de la Semana Nacional del Tránsito. En abril de 1974 recibió una carta de felicitación del Ministro de las FAR, de Antonio Pérez Herrero y Blas Roca Calderío con motivo de la emisión del número 500 de la publicación. En 1977 se le otorgó la Bandera de Honor de la UJC. En 1980 recibió un diploma de la FMC por los 20 años de divulgación de los objetivos y tareas de esta organización.

Capítulo 1. Fundamentación teórica

1.2.3 Tendencias actuales en la publicación de portales Web

Al realizar un estudio de las diferentes revistas y portales Web en Cuba y en el mundo se obtuvieron tendencias reflejadas en cada uno de ellos, las que se tendrán presentes para el desarrollo del portal Web de la revista Pionero; entre ellas se encuentran:

Compartir diferentes contenidos: Permite compartir documentos, videos, sonidos, imágenes o solamente la información que se está buscando.

Foros: Permite al usuario realizar comentarios y dar opiniones acerca de cualquier tema tratado en el portal.

Publicación de noticias: Permite a los usuarios la búsqueda de información en el portal sobre temas de adolescentes y pioneros.

Descargas de archivos: Permite a los usuarios la descarga de los contenidos compartidos en el portal.

Buscador: Permite a los usuarios de forma rápida y ágil encontrar los contenidos y las noticias publicadas en el portal.

Subscripción a boletín con artículos de la revista: Permite que el usuario tenga un resumen de las informaciones más actualizadas de la página siendo enviado a su correo particular.

Libro de visitas: Permite llevar un control de las visitas realizadas al portal de la revista.

1.3 Sistema de Gestión de Contenidos (CMS)

Un Sistema de Gestión de Contenidos es una herramienta que permite a un editor crear, clasificar y publicar cualquier tipo de información en una página Web. Generalmente los CMS hacen uso de una base de datos, de modo que el editor simplemente actualiza una base de datos, incluyendo nueva información o editando la existente. [6]

Capítulo 1. Fundamentación teórica

1.3.1 CMS Joomla

Es un sistema de administración de contenidos de código abierto construido con el lenguaje PHP bajo la licencia GPL. Realiza publicaciones utilizando una base de datos *MySQL*. Presenta una larguísima lista de extensiones que se pueden mejorar y lograr que se ajusten a las necesidades del usuario. [7]

Ventajas del CMS Joomla

- Organización del sitio Web: Está preparado para organizar eficientemente los contenidos de su portal en secciones y categorías, lo que facilita la navegabilidad para los usuarios y permite crear una estructura sólida, ordenada y sencilla para los administradores. Desde el panel administrador de Joomla usted podrá crear, editar y borrar las secciones y categorías de su sitio de la manera en que más le convenga.
- Publicación de Contenidos: Podrá crear páginas ilimitadas y editarlas desde un sencillo editor que permite formatear los textos con los estilos e imágenes deseados. Los contenidos son totalmente editables y modificables.
- Escalabilidad e implementación de nuevas funcionalidades: Ofrece la posibilidad de instalar, desinstalar y administrar componentes y módulos, que agregarán servicios de valor a los visitantes de su sitio Web, por ejemplo: galerías de imágenes, foros, *newsletters*, clasificados, etc.
- Administración de usuarios: Permite almacenar datos de usuarios registrados y también la posibilidad de enviar E-mails masivos a todos los usuarios.
- La administración de usuarios es jerárquica, y los distintos grupos de usuarios poseen diferentes niveles de facultades/permisos dentro de la gestión y administración del sitio.
- Diseño y aspecto estético del portal: Es posible cambiar todo el aspecto del portal Web tan solo con un par de *click*, gracias al sistema de *templates* que utiliza Joomla.
- Navegación y menú: Totalmente editables desde el panel administrador de Joomla.
- Administrador de Imágenes: Posee una utilidad para subir imágenes al servidor y usarlas en todo el portal.
- Disposición de módulos modificable: En un sitio creado con Joomla, la posición de módulos puede acomodarse como se prefiera.
- Encuestas: Posee un sistema de votaciones y encuestas dinámicas con resultados en barras porcentuales.

Capítulo 1. Fundamentación teórica

- *Feed* de Noticias: Trae incorporado un sistema de sindicación de noticias por RSS/XMS de generación automática.
- Publicidad: Es posible hacer publicidad en el sitio usando el Administrador de *Banners*.
- Estadísticas de visitas: Con información de navegador, OS, y detalles de los documentos (páginas) más vistos. [7]

Desventaja del CMS Joomla

- Insuficiente interfaz administrativa.
- Limitaciones en la personalización de rangos de usuarios.
- Dependencia excesiva del *JavaScript* en su panel de administración.
- Afectaciones de rendimiento al procesar mucha información. [7]

1.3.2 CMS Drupal

Drupal: Proviene del holandés “*druppel*” que significa “gota”. Es un sistema de gestión de contenido que brinda muchas facilidades a la hora de confeccionar portales Web, pues cuenta con elementos como un sistema de gestión de usuarios simple, algo crucial en un portal que se espera sea visitado por una gran cantidad de personas. También facilita la creación de foros y áreas de comentarios, que son factores importantes a la hora del intercambio constante de opiniones que se generará entre los visitantes al sitio, sobre los distintos temas que se publiquen. Se le suma a esto el sistema de agrupación por taxonomías, que hace más sencillo la agrupación de datos como los gustos y preferencias de cada usuario, dado que será útil conocer sobre qué temas le interesa a cada uno informarse. Además, está basado en código abierto, con licencia GNU/GPL, escrito en PHP, desarrollado y mantenido por una activa comunidad de usuarios, que logran mantener una buena arquitectura de programación concentrando las principales características de los CMS en el núcleo del sistema. Algunos de los gestores de base de datos que soporta son *MySQL* y *PostgreSQL*. Es adecuado para la realización de portales Web al poseer una flexibilidad y adaptabilidad permitiendo que los usuarios que lo utilicen puedan realizarle configuraciones a los módulos, inclusive poder añadirle algunos desarrollados a la medida según las necesidades del portal a realizar. [8]

Características del CMS Drupal

Ayuda on-line

Capítulo 1. Fundamentación teórica

Un robusto sistema de ayuda online y páginas de ayuda para los módulos del “núcleo”, tanto para usuarios como para administradores.

Búsqueda

Todo el contenido en Drupal es totalmente indexado en tiempo real y puede ser consultado en cualquier momento.

Código abierto

El código fuente de Drupal está libremente disponible bajo los términos de la licencia GNU/GPL. Al contrario que otros sistemas de “*blogs*” o de gestión de contenido propietarios, es posible extender o adaptar Drupal según las necesidades particulares.

Módulos

La comunidad de Drupal ha contribuido muchos módulos que proporcionan funcionalidades como “página de categorías”, autenticación mediante *jabber*, mensajes privados, *bookmarks*, y otros.

Personalización

Un robusto entorno de personalización está implementado en el núcleo de Drupal. Tanto el contenido como la presentación pueden ser individualizados de acuerdo las preferencias definidas por el usuario.

URLs amigables

Drupal usa el *mod_rewrite* de Apache para crear URLs que son manejables por los usuarios y los motores de búsqueda. [9]

Después del estudio de los principales CMS de código abierto se decide emplear el CMS Drupal en su versión 7.21 debido a las facilidades que brinda, que aunque tiene algunas desventajas relacionadas con su complejidad de uso, estas no impiden la realización correcta del portal Web que se desea desarrollar. Es un sistema de gestión de contenido modular muy configurable que permite publicar artículos, imágenes, u otros archivos y servicios añadidos como foros, encuestas, votaciones, blogs y administración de usuarios y permisos. Drupal es un sistema dinámico: en lugar de almacenar sus contenidos en archivos

Capítulo 1. Fundamentación teórica

estáticos en el sistema de ficheros del servidor de forma fija, el contenido textual de las páginas y otras configuraciones son almacenados en una base de datos y se editan utilizando un entorno Web.

1.4 Lenguajes empleados

1.4.1 Lenguaje Unificado de Modelado 2.0 (UML)

La metodología escogida para el desarrollo de la solución propuesta, utiliza *Unified Modeling Language* (**UML**) como herramienta, este permite visualizar las reglas de creación, la estructura y el comportamiento de objetos, procesos y artefactos surgidos durante las fases de creación del producto final en el proceso de desarrollo del software.

Es una notación donde se construyen modelos de sistemas por medio de conceptos orientados a objetos. Posee un conjunto de notaciones, diagramas estándares y describe la semántica esencial de lo que estos símbolos significan. [10]

1.4.2 Lenguaje de Marcado de Hipertexto 4.0 (HTML)

HyperText Markup Language (por sus siglas en ingles **HTML**): Es el Lenguaje de Marcas de Hipertextos diseñado para estructurar textos y presentarlos en forma de hipertexto, que es el formato estándar de las páginas Web. Gracias a los navegadores como: Internet Explorer, Opera, Firefox, Netscape y Safari, el HTML se ha convertido en uno de los formatos más populares y fáciles de aprender que existen para la elaboración de documentos para la Web. Este lenguaje permite la introducción de referencias a otras páginas por medio de los enlaces de hipertexto y crear lenguajes de codificación descriptivos. Define una estructura de documentos jerárquica, con elementos y componentes interconectados. No tiene un conjunto implícito de convenciones de señalización. Soporta, por tanto, un conjunto flexible de juegos de etiquetas. [8]

1.4.3 Hojas de Estilos en Cascada (CSS)

Cascading Style Sheets (por sus siglas en ingles **CSS**): Es un lenguaje de hojas de estilos creado para controlar el aspecto o presentación de los documentos electrónicos definidos con HTML y XHTML. CSS es una forma de separar los contenidos y su presentación, es de invaluable ayuda a la hora de crear páginas Web de estructura compleja. [11]

Capítulo 1. Fundamentación teórica

Separar la definición de los contenidos y la definición de su aspecto presenta numerosas ventajas:

- Obliga a crear documentos HTML/XHTML bien definidos y con significado completo.
- Mejora la accesibilidad del documento.
- Reduce la complejidad de su mantenimiento.
- Permite visualizar el mismo documento en dispositivos diferentes. [11]

Al crear una página Web, se utiliza en primer lugar el lenguaje HTML/XHTML para marcar los contenidos, es decir, para designar la función de cada elemento dentro de la página: párrafo, titular, texto destacado, tabla, lista de elementos, etc.

Una vez creados los contenidos, se utiliza el lenguaje **CSS** para definir el aspecto de cada elemento: color, tamaño y tipo de letra del texto, separación horizontal y vertical entre elementos, posición de cada elemento dentro de la página, etc.

1.4.4 Lenguaje de Programación PHP 5.0

PHP es un lenguaje de programación generalmente usado en la programación de sitios Web dinámicos y actualmente es el lenguaje de desarrollo de sitios más usado en todo el mundo. Fue originalmente creado por Rasmus Lerdorf para presentar su portafolio de trabajo y otras cosas en el año de 1994. Originalmente fue desarrollado en *Perl*. [12]

Características fundamentales de PHP:

- Es un lenguaje del lado del servidor, por lo que los script se ejecutan remotamente y el resultado aparece en el ordenador.
- Tiene soporte para muchos tipos de bases de datos, entre las principales están *MySQL*, *PostgreSQL*, *SQLite*, entre otras.
- Es embebido en código HTML.
- PHP soporta ambos paradigmas de programación: estructurada y orientada a objetos. [12]

Capítulo 1. Fundamentación teórica

1.4.5 Lenguaje de Programación *JavaScript*

JavaScript es un lenguaje compacto, y basado en objetos, diseñado para el desarrollo de aplicaciones cliente-servidor a través de Internet. En una aplicación cliente para un navegador, las sentencias **JavaScript** pueden reconocer y responder a eventos generados por el usuario, como hacer *click* con el mouse, información en formularios y navegación de documento a documento. Por ejemplo, se puede escribir una función **JavaScript** que verifique que la información ingresada por el usuario sea correcta sin que haya transmisión de datos por la red. [13]

Con **JavaScript** es posible validar los elementos antes de que el usuario se los envíe a un servidor. De esta forma se reduce la cantidad de transacciones que se efectúan a través del protocolo http y las posibilidades de que se genere un error durante la inserción de datos. **JavaScript** también puede leer y escribir *cookies*, una operación que hasta ahora únicamente podía desarrollar el servidor Web. [14]

1.5 Entorno de Desarrollo Integrado (IDE)

Un IDE es un entorno de programación que ha sido empaquetado como un programa de aplicación; es decir, consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica (GUI). Los IDEs pueden ser aplicaciones por sí solas o pueden ser parte de aplicaciones existentes. [15]

Es importante una correcta selección del IDE de programación debido que este proporciona un mejor aprovechamiento del tiempo y facilidad de uso.

1.5.1 NetBeans 7.1

Es un conjunto de herramientas utilizadas por los programadores, que incluye por lo general, un buen editor de código, administrador de proyectos y archivos, enlace a compiladores e integración con sistemas controladores de versiones o repositorios, además de brindar facilidades para la construcción de interfaces gráficas de usuario. Su licencia permite construir tantas aplicaciones *Open Source* como comerciales. Los proyectos desarrollados poseen lanzadores (*launchers*) para cada plataforma. Hace fuerte hincapié sobre la construcción del software de forma modular, módulo sobre módulo, ofreciéndonos implementados los mecanismos de descubrimiento de nuevos módulos (y de actualizaciones de los existentes) desde repositorios remotos, resolución de dependencias, activación/desactivación de módulos en caliente y

Capítulo 1. Fundamentación teórica

comunicación entre los mismos. Cuenta con módulos que hacen extensible su uso hacia otros lenguajes de programación como PHP, siendo compatible con la versión 5.3 de PHP. Tiene una creciente comunidad de usuarios, pues permite la depuración y ejecución de programas escritos. Teniendo en cuenta que se utiliza el CMS Drupal para el desarrollo del Portal de Aplicaciones Educativas (PAE), es importante destacar que permite la creación de módulos para el CMS utilizado, brindando autocompletamiento. [15]

1.5.2 Apache Web Server 2.2.8

El servidor *Apache* es un servidor Web HTTP de código abierto para plataformas Unix (BSD, GNU/Linux, etc.), *Windows*, *Macintosh* y otras, que implementa el protocolo HTTP/1.1 y la noción de sitio virtual. El servidor Apache se desarrolla dentro del proyecto HTTP Server (httpd) de la *Apache Software Foundation*. Es altamente configurable, admite bases de datos de autenticación y negociado de contenido, aunque carece de una interfaz gráfica que ayude en su configuración. *Apache* es una aplicación que permite montar un servidor Web en cualquier equipo y casi cualquier sistema operativo. Al contrario que IIS (*Internet Information Server*) que sólo funciona en sistemas operativos de *Microsoft*. Con los módulos adecuados, *Apache* puede soportar también ASP. [16]

Entre sus características se destacan:

- Multiplataforma: Tiene la capacidad de funcionar en más de un sistema operativo con similares características y sin que su funcionalidad varíe en exceso.
- Modular: Puede ser adaptado a diferentes entornos y necesidades, con los diferentes módulos de apoyo que proporciona, y con las API de programación de módulos.
- Extensible: Por su propiedad de ser modular se han desarrollado diversas extensiones entre las que destaca PHP, un lenguaje de programación del lado del servidor. [16]

1.6 Sistemas Gestores de Base de Datos

Un Sistema Gestor de Bases de Datos (SGBD) es una colección de programas cuyo objetivo es servir de interfaz entre la base de datos, el usuario y las aplicaciones. Se compone de un lenguaje de definición de datos, de un lenguaje de manipulación de datos y de un lenguaje de consulta. Un SGBD permite definir los

Capítulo 1. Fundamentación teórica

datos a distintos niveles de abstracción y los manipula, garantizando la seguridad e integridad de los mismos. [17]

1.6.1 MySQL

Es un gestor de base de datos sencillo de usar e increíblemente rápido. También es uno de los motores de base de datos más usados en Internet, la principal razón de esto es que es gratis para aplicaciones no comerciales. [18]

Gran parte de las viejas bases de datos presentan problemas por utilizar sistemas obsoletos, lo que complica innecesariamente las tareas de administración. Las herramientas de *MySQL* son potentes y flexibles, sin sacrificar su capacidad de uso. [18]

Características de *MySQL*

- Aprovecha la potencia de sistemas multiprocesador, gracias a su implementación multihilo.
- Soporta gran cantidad de tipos de datos para las columnas.
- Portabilidad entre sistemas.
- Soporta hasta 32 índices por tabla.
- Gestión de usuarios y contraseñas manteniendo un muy buen nivel de seguridad en los datos.
- Base de datos relacional. [18]

1.6.2 PostgreSQL 8.4

PostgreSQL: Es un sistema gestor de bases de datos relacionales de código abierto. Cuenta con más de 15 años de desarrollo activo y una arquitectura probada que se ha ganado una sólida reputación de fiabilidad, integridad de datos y la corrección de los mismos. Es multiplataforma, ejecutándose en los principales sistemas operativos. Soporta almacenamiento de objetos binarios grandes, como imágenes, sonidos o vídeo. Es altamente escalable, tanto en la enorme cantidad de datos que puede manejar y en el número de usuarios concurrentes que puede acomodar. Por sus características, garantiza la integridad de los datos, la velocidad de acceso y consultas a la base de datos. Incorpora una estructura de datos array y permite la declaración de funciones propias. Permite la gestión de diferentes usuarios, además de los permisos asignados a cada uno. [19]

Capítulo 1. Fundamentación teórica

Características de *PostgreSQL*

- Múltiples métodos de autenticación.
- Disponible para Linux, UNIX en todas sus variantes y *Windows* 32/64bit. [20]

El sistema gestor de base de datos seleccionado fue *PostgreSQL* 8.4, puesto que su configuración por defecto fue diseñada para ejecutarse en sistemas con poca memoria. Su ventaja en velocidad con respecto a *MySQL* se puede evidenciar drásticamente en un ambiente de grandes procesadores *multicore*.

1.7 Metodología a utilizar

Las metodologías de desarrollo de software son un conjunto de procedimientos, técnicas y ayudas a la documentación para el desarrollo de productos de software. Pueden ser comparadas con un plan de contingencias en el que se va indicando paso a paso todas las actividades a realizar para lograr el producto informático deseado, indicando además quiénes deben participar en el desarrollo de las actividades y qué papel deben tener. Detallan la información que se debe producir como resultado de una actividad y la información necesaria para comenzar la próxima actividad. [21]

Las Metodologías Ágiles o “ligeras” constituyen un nuevo enfoque en el desarrollo de software, mejor aceptado por los desarrolladores de *e-projects* que las metodologías convencionales (ISO-9000, CMM, etc) debido a la simplicidad de sus reglas y prácticas, su orientación a equipos de desarrollo de pequeño tamaño, su flexibilidad ante los cambios y su ideología de colaboración. [21]

1.7.1 eXtremeProgramming (XP)

Es una metodología reciente que se utiliza en los procesos de desarrollo de software. Posee como filosofía, satisfacer por completo las necesidades del cliente, al cual incorpora dentro del equipo. Está diseñada para grupos pequeños de programadores (más de 10 sería muy complicado) y surge como respuesta o posible solución a los problemas que se derivan de los cambios de requerimiento. Se plantea como una metodología a emplear en proyectos de riesgos para aumentar la productividad. XP se basa en la simplicidad, la comunicación y la retroalimentación o reutilización del código desarrollado. Existen algunas críticas; se piensa que solo funciona con programadores muy buenos, capaces de hacer un buen diseño, sencillo y fácilmente extensible. [21]

Capítulo 1.Fundamentación teórica

1.7.2 Scrum

Define un marco para la gestión de proyectos, que se ha utilizado con éxito durante los últimos 10 años. Sus principales características se pueden resumir en dos: el desarrollo de software se realiza mediante iteraciones, denominadas *sprints*, con una duración de 30 días. El resultado de cada *sprint* es un incremento ejecutable que se muestra al cliente. La segunda característica importante son las reuniones a lo largo del proyecto, entre ellas destaca la reunión diaria de 15 minutos del equipo de desarrollo para coordinación e integración. [21]

1.7.3 Metodología de Desarrollo de Software OpenUP

Características de OpenUP

1. Preserva la esencia del *Rational Unified Process*(RUP):
 - Desarrollo iterativo e incremental.
 - Desarrollo dirigido por Casos de Uso.
2. Centrado en la Arquitectura.
3. La mayoría de las características opcionales de RUP se excluyeron y otras se unificaron para simplificar el proceso: sólo lo fundamental está incluido, sin dejar de ser completo y extensible (alrededor de 20 artefactos).
4. Está pensado para proyectos de 3 a 6 personas, 3 a 6 meses de esfuerzo de programación.
5. Es un proceso iterativo para el desarrollo de software.
 - Mínimo: Solo incluye el contenido del proceso fundamental.
 - Completo: Puede ser manifestado como proceso entero para construir un sistema.
 - Extensible: Puede ser utilizado como base para agregar o para adaptar más procesos. [21]

Para el desarrollo del proyecto se determinó como metodología OpenUP “por ser un proceso modelo y extensible, dirigido a la gestión y desarrollo de proyectos de software basados en desarrollo iterativo, ágil e incremental apropiado para proyectos pequeños y de bajos recursos; y es aplicable a un conjunto amplio de plataformas y aplicaciones de desarrollo.” [21]

1.8 Herramientas Utilizadas

1.8.1 Pgadmin III

Es una aplicación gráfica para administrar el gestor de bases de datos *PostgreSQL*. Está escrita en C++ usando la librería gráfica multiplataforma *wxWidgets*, lo que permite que se pueda usar en Linux, *FreeBSD*, *Solaris*, *Mac OS X* y *Windows*. Es capaz de gestionar versiones a partir de *PostgreSQL* 7.3 ejecutándose en cualquier plataforma, así como versiones comerciales de *PostgreSQL* como *PervasivePostgres*, *EnterpriseDB*, *MammothReplicator* y *SRA PowerGres*. Está diseñado para responder a las necesidades de todos los usuarios, desde escribir consultas SQL simples hasta desarrollar bases de datos complejas. La aplicación también incluye un editor SQL con resaltado de sintaxis, un editor de código de la parte del servidor, un agente para lanzar scripts programados. La conexión al servidor puede hacerse mediante conexión TCP/IP o *Unix Domain Sockets* y puede encriptarse mediante SSL para mayor seguridad. [8]

1.8.2 Visual Paradigm para UML 8.0

Visual Paradigm para UML es una de las llamadas herramientas CASE o *Computer Assisted Software Engineering*, en español, Ingeniería de Software Asistida por Computadora. Facilita el modelado de artefactos en un Proceso de Desarrollo de Software, mediante el lenguaje de modelado UML. Soporta Ingeniería Inversa, generación de código, posee generador de informes, exporta diagramas en diversos formatos entre otras funciones importantes. Permite diseñar el negocio y el sistema del presente trabajo dando una mayor precisión y permitiendo obtener resultados exitosos. Visual Paradigm para UML provee de diversas funcionalidades de modelado de Casos de Uso, Flujos de Eventos, Cuadrícula para Actor y generación de Diagrama de Actividades. Visual Paradigm para UML produce documentación del sistema en formato PDF, HTML y de MS Word. Los desarrolladores pueden diseñar la documentación del sistema con una plantilla de diseño. [22]

Conclusiones parciales

La realización del estudio de diferentes sistemas homólogos permitió caracterizar las tendencias actuales en el desarrollo de portales Web, dando lugar al diseño de las funcionalidades y servicios a incluir en el portal Web de la revista Pionero. Se seleccionó la metodología para el desarrollo de software, así como las herramientas y lenguajes necesarios para la correcta implementación del portal Web de la revista Pionero

Capítulo 1. Fundamentación teórica

haciendo uso de las potencialidades de la Web 2.0 fomentando la distribución y el intercambio de la información entre los usuarios.

Capítulo 2. Características del Sistema

En este capítulo se realizará un estudio de los requisitos funcionales y no funcionales que deberá presentar el sistema, así como identificar y describir los artefactos que serán creados para guiar el diseño y la posterior implementación de la aplicación. Además se realizará una breve descripción del sistema propuesto para la siguiente investigación. Como pieza fundamental de modelaje se usará el lenguaje unificado de modelado UML 2.0 y como herramienta CASE el Visual Paradigm 8.0 Enterprise Edition.

2.1 Modelo de Dominio

El modelo de dominio puede utilizarse para capturar y expresar el entendimiento ganado en un área bajo análisis como paso previo al diseño de un sistema, ya sea de software o de otro tipo. Similares a los mapas mentales utilizados en el aprendizaje, el modelo de dominio es utilizado por el analista como un medio para comprender el sector industrial o de negocios al cual el sistema va a servir. [23]

2.1.1 Diagrama de Clases del Modelo del Dominio

Para entender la estructura organizacional que se modela se realiza el siguiente mapa conceptual. El mismo relaciona, clases usando el lenguaje UML para observar las relaciones entre entidades, actores y comprender el flujo de trabajo que se evidencia en el negocio que se modela.

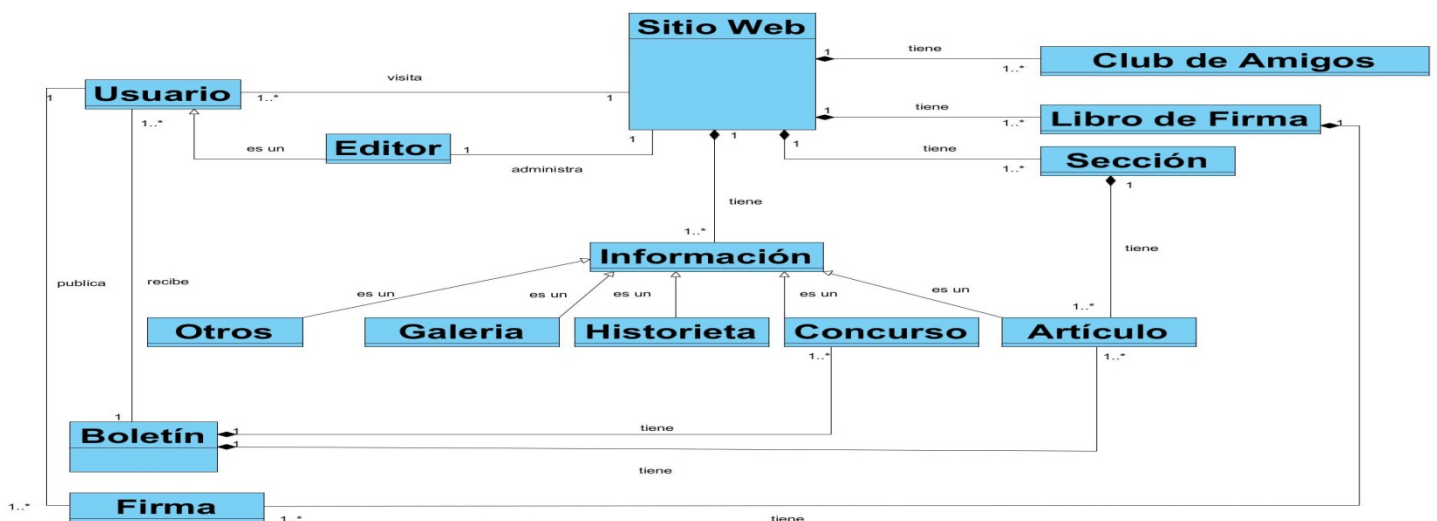


Figura 2. 1: Diagrama de Clases del Modelo del Dominio.

Capítulo 2. Características del Sistema

Descripción de Clases del Modelo del Dominio

Usuario: Es el usuario genérico que accede a información o servicios.

Editor: Usuario encargado de publicar la información y los servicios que brinda el sitio Web.

Lector: Usuario que accede al sitio Web en busca de información.

Información: Es cualquier artefacto de información generado por el editor y consultado por los lectores.

Sección: Secciones de la revista a las cuales pertenecen los artículos que se publican.

Artículo: Pieza básica de las publicaciones en la revista.

Concurso: Vista para los usuarios que muestra los datos de los concursos actuales.

Historieta: Vista de historietas publicadas por la revista.

Galería: El sitio brinda la posibilidad de revisar galerías de imágenes.

Firma: Comentarios, opiniones o mensajes que los usuarios dejan en el libro de visitas.

Libro de Firma: Donde se guardan todas las firmas dejadas por los usuarios en el libro de visitas.

Sitio Web: Es la revista donde el editor genera la información y el lector la consulta.

Otros: Corresponde con informaciones varias que se publican en el Sitio Web como por ejemplo, recetas de cocina, vínculos a pequeños sitios relacionados con el ahorro, el cambio climático y otras temáticas.

2.2 Especificación de los requisitos del sistema

En el siguiente apartado se listan los requisitos funcionales y no funcionales que fueron identificados durante el estudio inicial de la solución que se propone.

Los requisitos funcionales son condiciones o capacidades que se necesita para resolver un problema y lograr un objetivo determinado desde el punto de vista del usuario.

Capítulo 2. Características del Sistema

A continuación se listan los requisitos funcionales con prioridad Alta.

2.2.1 Requisitos Funcionales

No.	Nombre	Descripción	Prioridad para el cliente	Complejidad
RF 1	Crear usuario	Se requiere los campos nombre completo, edad, grado escolar, sexo y provincia.	Alta	Alta
RF 2	Editar usuario	Se pueden modificar los campos nombre completo, edad, grado escolar, sexo y provincia.	Alta	Alta
RF 3	Eliminar usuario	Se puede cancelar la cuenta de usuario, eliminando la cuenta y su contenido.	Alta	Alta
RF 4	Autenticar usuario	Se requiere los campos usuario y contraseña. Si son correctos el sistema permite acceder a las funcionalidades que requieren autenticación, de lo contrario se muestra un mensaje "Usuario o contraseña incorrectos".	Alta	Alta
RF 5	Listar usuarios	Se muestran los campos nombre completo, edad, grado escolar, sexo y provincia.	Alta	Alta
RF 6	Mostrar Perfil de Usuario	Se muestran los campos nombre completo, edad, grado escolar, sexo y provincia.	Alta	Alta

Capítulo 2. Características del Sistema

RF 7	Asignar Rol a Usuario	Se le asigna un rol al usuario de acuerdo a las funcionalidades a las cuales tenga acceso.	Alta	Alta
RF 8	Crear Sección	Permite crear una sección para publicar diferentes tipos de artículos.	Alta	Alta
RF 9	Editar Sección	Permite editar una sección para publicar diferentes tipos de artículos.	Alta	Alta
RF 10	Eliminar Sección	Permite eliminar una sección ya creada.	Alta	Alta
RF 11	Listar Secciones	Se muestran todas las secciones creadas en el portal Web.	Alta	Alta
RF 12	Crear Artículo	Es la pieza básica de las publicaciones en la revista, se requiere los campos: publicado, autor, sección, observación. Además opcionalmente pueden especificarse los campos: imagen, correo electrónico e imágenes que el crédito de la fuente.	Alta	Alta
RF 13	Editar Artículo	Se pueden modificar los campos: publicado, autor, sección, observación. Además si existen pueden modificarse los campos: imagen, correo electrónico e imágenes que el crédito de la fuente.	Alta	Alta

Capítulo 2. Características del Sistema

RF 14	Eliminar Artículo	Se puede eliminar un artículo ya creado.	Alta	Alta
RF 15	Listar Artículos	Se muestran todos los artículos existentes en la revista.	Alta	Alta
RF 16	Crear Concurso	Se requiere los campos nombre, tipo, fecha cierre y bases.	Alta	Alta
RF 17	Editar Concurso	Se pueden modificar los campos nombre, tipo, fecha cierre y bases.	Alta	Alta
RF 18	Eliminar Concurso	Se puede eliminar un concurso ya creado.	Alta	Alta
RF 19	Listar Concursos	Se muestran todos los concursos existentes en la revista.	Alta	Alta
RF 20	Solicitar Inscripción	Se requiere los campos nombre completo, provincia, descripción, email, edad y grado escolar.	Alta	Alta
RF 21	Editar Inscripción	Se pueden modificar los campos nombre completo, provincia, descripción, email, edad y grado escolar. Además el editor puede o no aprobar la inscripción.	Alta	Alta

Capítulo 2. Características del Sistema

RF 22	Eliminar Inscripción	Se puede eliminar una inscripción ya creada.	Alta	Alta
RF 23	Listar Inscripciones	Se listan todas las inscripciones existentes en la revista.	Alta	Alta
RF 24	Crear Firma	Se requiere los campos nombre completo y texto. Además opcionalmente puede especificarse el campo correo electrónico.	Alta	Alta
RF 25	Editar Firma	Se pueden modificar los campos nombre completo y texto. Además si existe puede modificarse el campo correo electrónico. El editor puede o no aprobar la firma.	Alta	Alta
RF 26	Eliminar Firma	Se puede eliminar una firma ya creada.	Alta	Alta
RF 27	Listar Peticiones de Firma	Se listan todas las firmas existentes en la revista.	Alta	Alta
RF 28	Crear Historieta	Se requiere de los campos nombre, ilustrador y dibujos que será de elementos jpg, bmp o gif.	Alta	Alta
RF 29	Editar Historieta	Se pueden modificar los campos nombre, ilustrador y dibujos que será de elementos jpg, bmp o gif.	Alta	Alta

Capítulo 2. Características del Sistema

RF 30	Eliminar Historieta	Se puede eliminar una historieta ya creada.	Alta	Alta
RF 31	Listar Historietas	Se pueden listar las historietas existentes en la revista.	Alta	Alta
RF 32	Crear Galería	Se requiere de los campos nombre, fecha, fotógrafo y descripción.	Alta	Alta
RF 33	Editar Galería	Se pueden modificar los campos nombre, fecha, fotógrafo y descripción.	Alta	Alta
RF 34	Eliminar Galería	Se puede eliminar una galería ya creada.	Alta	Alta
RF 35	Listar Galerías	Se pueden listar las galerías ya existentes.	Alta	Alta
RF 36	Crear Carta	Se requiere los campos nombre completo, email, fecha y texto.	Alta	Alta
RF 37	Editar Carta	Se pueden modificar Los campos, nombre completo, email, fecha y texto. Además editor puede o no aprobar la carta.	Alta	Alta
RF 38	Eliminar Carta	Se puede eliminar una carta ya creada antes de ser aprobada.	Alta	Alta

Capítulo 2. Características del Sistema

RF 39	Listar Cartas	Se pueden listar todas las cartas ya existentes en la revista.	Alta	Alta
RF 40	Crear Encuesta	Permite crear una encuesta para los usuarios.	Alta	Alta
RF 41	Editar Encuesta	Permite modificar una encuesta ya creada.	Alta	Alta
RF 42	Eliminar Encuesta	Permite eliminar una encuesta ya creada.	Alta	Alta
RF 43	Listar Encuestas	Se pueden listar las encuestas ya existentes en la revista.	Alta	Alta
RF 44	Mostrar Galería	Se puede mostrar la galería que se desee.	Alta	Alta
RF 45	Búsqueda Simple	Permite realizar una búsqueda sencilla de los contenidos publicados en la revista.	Alta	Alta
RF 46	Búsqueda Avanzada	Permite realizar una búsqueda avanzada de los contenidos publicados en la revista.	Alta	Alta
RF 47	Enviar Boletín	Permite al Editor enviar un boletín por correo a los usuarios.	Alta	Alta

Capítulo 2. Características del Sistema

RF 48	Crear Receta	Se requiere los campos nombre y preparación	Alta	Alta
RF 49	Editar Receta	Se pueden modificar los campos nombre y preparación. Además el Editor puede o no aprobar la receta enviada por el usuario.	Alta	Alta
RF 50	Eliminar Receta	Se puede eliminar una receta ya creada.	Alta	Alta
RF 51	Listar Recetas	Se pueden listar todas las recetas existentes en la revista.	Alta	Alta

Tabla 2. 1: Listado de Requisitos funcionales.

2.2.2 Requisitos No Funcionales

Los requisitos no funcionales especifican criterios que pueden usarse para juzgar la operación de un sistema en lugar de sus comportamientos específicos. Se refieren a todos los requisitos que ni describen información a guardar, ni funciones a realizar; en otras palabras son las propiedades o cualidades que el producto debe tener. [24]

Usabilidad:

- Diseño limpio y claro que facilite la navegación.
- Estructura simple y lo más intuitiva posible de la arquitectura de la información.
- Diseño ajustable a la resolución de la pantalla.

Confiabilidad:

Capítulo 2. Características del Sistema

- Se asignarán los permisos de acceso, escritura, lectura en dependencia del rol que desempeñe cada usuario del sistema.
- El sistema debe permitir usuarios con diferentes niveles de acceso.
- La información manejada por el sistema estará protegida de acceso no autorizado.
- Llevar un registro de sucesos donde se archiven los eventos del sistema incluyendo los eventos de error, inicio de sesión, cierre de sesión y modificación de la información.
- En caso de que el sistema presente alguna falla, los errores se deben mostrar sin detalles de información que pueda comprometer la seguridad e integridad del mismo.
- La información manejada por el sistema será objeto de cuidadosa protección contra la corrupción y estados inconsistentes.

Requisitos legales, de derecho de autor y otros:

- Uso de la licencia PHP License.
- Uso de la licencia Apache Software License (para versiones anteriores a 2.0).
- Uso de la licencia BSD de PostgreSQL.
- Uso de la licencia GNU/GPL para el CMS Drupal.

Requisitos de Hardware:

Cliente:

- 1 GB de memoria RAM
- Procesador Pentium IV

Servidor Web Apache:

- GB de memoria RAM

Capítulo 2. Características del Sistema

- 20 GB de disco duro
- Intel Core 2 Duo

Servidor base de datos:

- GB de memoria RAM
- GB de disco duro
- Intel Core 2 Duo

2.3 Descripción del sistema propuesto

El sistema propuesto consiste en un portal Web que contribuya a la divulgación y promoción en Internet de la revista Pionero. Esta aplicación debe cumplir los requisitos anteriormente descritos para satisfacer las necesidades de la institución y resolver las problemáticas planteadas en la introducción de esta investigación.

2.4 Descripción de Requisitos de Software

A continuación se describen los requisitos funcionales críticos con los que contará la aplicación.

Descripción del Requisito Crear Usuario

Número: 1	Nombre del requisito: Crear usuario
Programador: Agustin Medina Torrez.	Iteración Asignada: 1
Prioridad: Alta	Puntos Estimados: 0.1
Riesgo en Desarrollo: Riesgo Alto	Puntos Reales: 0.1
Descripción: El usuario debe completar los campos: nombre completo, edad, grado escolar, sexo y provincia, luego presiona el botón crear nueva cuenta y el sistema muestra un mensaje notificando la creación de la nueva cuenta y que la misma está pendiente de aprobación. En	

Capítulo 2. Características del Sistema

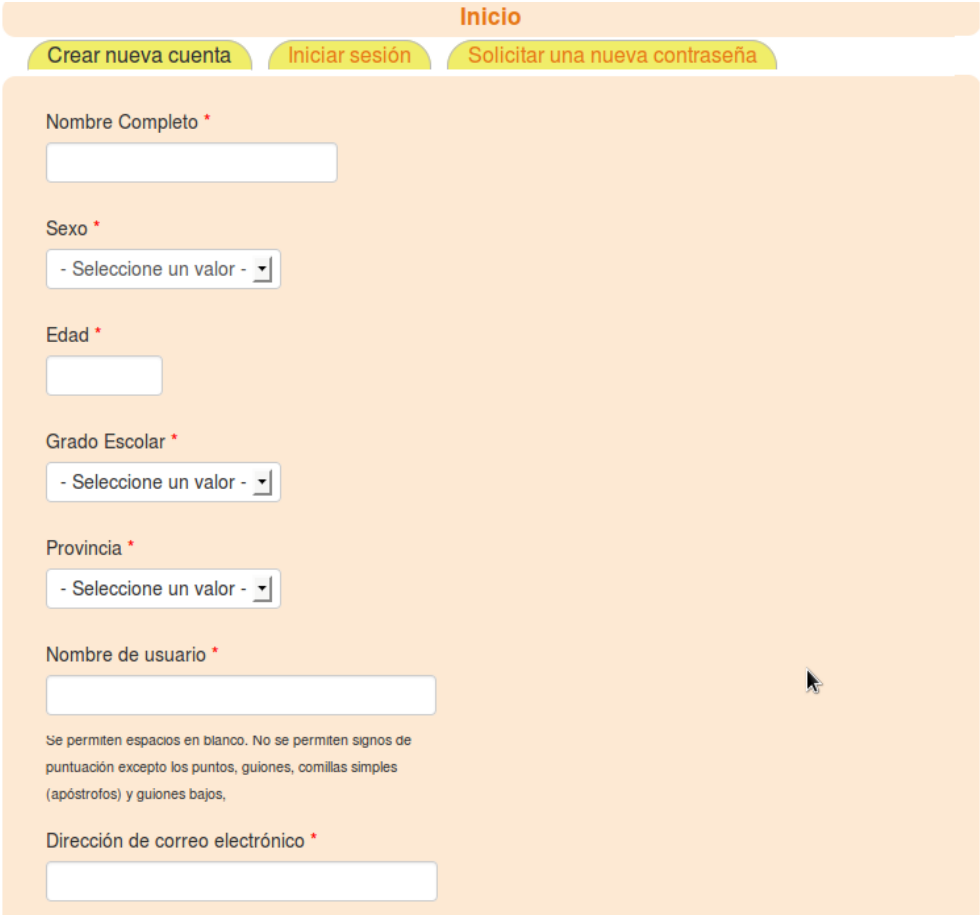
caso de completar incorrectamente un campo el sistema muestra un mensaje de error y señala el campo incorrecto.
Observaciones:
Prototipo de interfaz: 

Tabla 2. 2: Descripción del Requisito Crear Usuario.

Capítulo 2. Características del Sistema

Descripción del Requisito Crear Sección

Número: 2	Nombre del requisito: Crear sección
Programador: Osdany Toledo Reyes.	Iteración Asignada: 1
Prioridad: Alta	Puntos Estimados: 0.1
Riesgo en Desarrollo: Riesgo Alto	Puntos Reales: 0.1
Descripción: El sistema permite al usuario con el rol de editor crear una nueva sección referida a diferentes artículos de la revista.	
Observaciones:	
Prototipo de interfaz:	

Capítulo 2. Características del Sistema

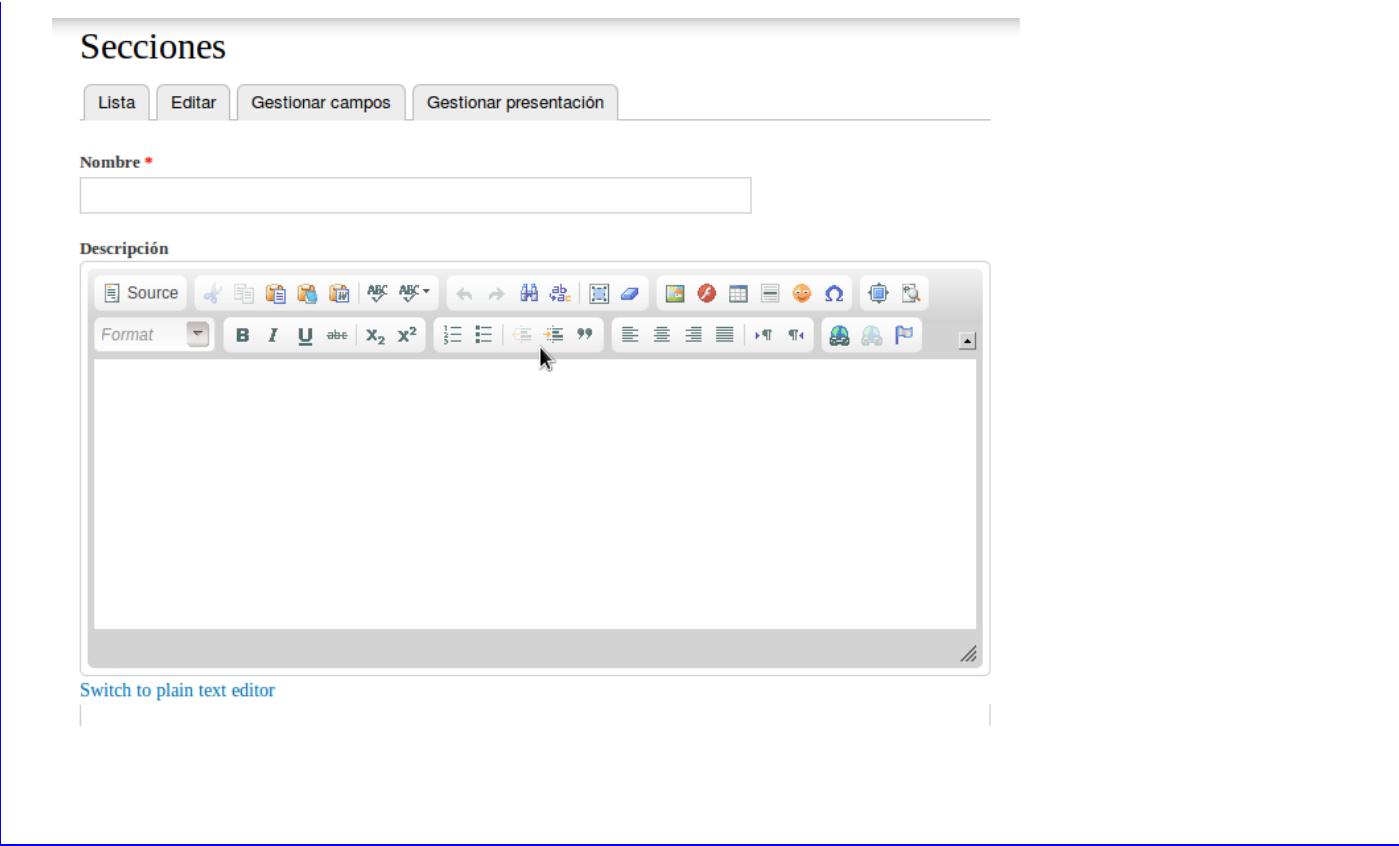


Tabla 2. 3: Descripción del Requisito Crear Sección

Descripción del Requisito Crear Artículo

Número: 3	Nombre del requisito: Crear artículo		
Programador: Agustin Medina Torrez.		Iteración Asignada: 1	
Prioridad: Alta		Puntos Estimados: 0.1	
Riesgo en Desarrollo: Riesgo Alto.		Puntos Reales: 0.1	
Descripción: El sistema permite al usuario con rol de editor la creación de un nuevo artículo completando los campos: publicado, autor y			

Capítulo 2. Características del Sistema

sección; opcionalmente pueden completarse los campos: imagen, correo electrónico e imágenes que es el crédito de la fuente. Luego presiona el botón crear nuevo artículo y el sistema muestra un mensaje notificando la creación del nuevo artículo, si un campo es completado incorrectamente el sistema muestra un mensaje de error y señala el campo incorrecto.

Observaciones:

Prototipo de interfaz:

Crear Artículo

Título *

Sección *

- Seleccione un valor -

Escoja la sección donde será ubicado el artículo, si no existe en la lista, añádala a través de la funcionalidad Adicionar Sección.

Por *

- Seleccione un valor -

Escoja el autor del artículo, si no existe en la lista, añádalo a través de la funcionalidad Adicionar Autor.

Imágenes *

Escoja el o los autores de las imágenes usadas en el artículo, si no existen en la lista, añádalos a través de la funcionalidad Adicionar Fotógrafo.

Fecha de publicación

Date

06/06/2013

E.g., 07/05/2013

Correo

Imagen principal *

Browse... Subir al servidor

Los archivos deben ser menores que 100 MB.

Tipos de archivo permitidos: png gif jpg jpeg.

Cuerpo * (Editar resumen)

Source Format B I U X₂ X²

Tabla 2. 4: Descripción del Requisito Crear Artículo.

Capítulo 2. Características del Sistema

Descripción del Requisito Crear Concurso

Número: 4	Nombre del requisito: Crear concurso
Programador: Osdany Toledo Reyes.	Iteración Asignada: 1
Prioridad: Alta	Puntos Estimados: 0.1
Riesgo en Desarrollo: Riesgo Alto.	Puntos Reales: 0.1
Descripción: El sistema permite al usuario con rol de editor la creación de un nuevo concurso completando los campos: nombre, tipo, fecha cierre y bases. Luego presiona el botón crear nuevo concurso y el sistema muestra un mensaje notificando la creación del nuevo concurso, si un campo es completado incorrectamente el sistema muestra un mensaje de error y señala el campo incorrecto.	
Observaciones:	
Prototipo de interfaz:	

Capítulo 2. Características del Sistema

Crear Concurso

Tipo *

- Selezionare un valor -

Nombre *

Fecha de Cierre

Date

06/06/2013

E.g., 07/06/2013

Bases *

Source

Format

B **I** **U** abc x_2 x^2

Tabla 2. 5: Descripción del Requisito Crear Concurso.

Descripción del Requisito Crear Firma

Número: 5	Nombre del requisito: Crear firma

Capítulo 2. Características del Sistema

Programador: Osdany Toledo Reyes.	Iteración Asignada: 1
Prioridad: Alta	Puntos Estimados: 0.1
Riesgo en Desarrollo: Riesgo Alto.	Puntos Reales: 0.1
Descripción: El sistema permite a los usuarios registrados la creación de una nueva firma completando los campos: nombre completo y texto; opcionalmente puede completarse el campo: correo electrónico. Luego presiona el botón crear nueva firma y el sistema muestra un mensaje notificando la creación de la nueva firma, si un campo es completado incorrectamente el sistema muestra un mensaje de error y señala el campo incorrecto.	
Observaciones:	
Prototipo de interfaz: El prototipo de interfaz para 'Crear Firma' presenta un formulario con un fondo naranja claro. En la parte superior, un encabezado naranja contiene el título 'Crear Firma' en texto naranja. El formulario está dividido en dos columnas. La columna izquierda contiene tres campos de texto con encabezados amarillos: 'Título *', 'Autor *' y 'Correo'. La columna derecha contiene un campo de texto grande con el encabezado 'Texto *'. En la parte inferior del formulario, hay dos botones amarillos: 'Guardar' y 'Vista previa'.	

Tabla 2. 6: Descripción del Requisito Crear Firma.

Conclusiones parciales

En este capítulo se obtuvo como resultado una vista general del sistema que sirviera para definir la estructura que deberían tener los elementos del sistema y los parámetros a tener en cuenta cuando se estuviera realizando el análisis con vistas a su implementación por parte de los desarrolladores.

Capítulo 3. Análisis y Diseño

3.1 Patrones de Arquitectura y Diseño

En este capítulo se diseña el sistema propuesto para que posteriormente sea implementado. Es en esta etapa constructiva donde se define la estructura de los datos, así como las clases, entidades y sus relaciones. Además se realiza el análisis para la construcción de los diferentes diagramas para lograr un correcto entendimiento de la aplicación a desarrollar.

3.1.1 Patrones de arquitectura

Al ser utilizado el CMS Drupal para el desarrollo del portal, la arquitectura y los patrones son heredados del mismo. El estilo arquitectónico de Drupal es un tema polémico en estos momentos, la afirmación de que Drupal presenta una arquitectura de n-capas, viene dada por la combinación de 2 elementos fundamentales: el primero siendo que esta arquitectura es en realidad un estilo de programación donde el objetivo principal es separar los diferentes aspectos del desarrollo, y permitir intercambiar porciones de la aplicación sin necesidad de modificar la aplicación completa [25] y el segundo, que la característica fundamental del CMS Drupal es su adaptabilidad mediante la adición de módulos, que son al final porciones de la aplicación que se modifican, además de dividir su contenido en una serie de elementos básicos: nodos (*nodes*), módulos (*modules*), bloques y *menus* (*blocks & menus*), permisos de usuarios (*user permissions*) y plantillas (*templates*). Los nodos son los elementos básicos en que Drupal almacena la información y los contenidos. Así a medida que el portal Web crece, aumenta también el número de nodos los cuales van formando un “depósito de nodos” cada vez mayor. Se puede decir que la primera capa de la estructura de Drupal la forma este “deposito” de nodos. Los módulos son los elementos que operan sobre los nodos y otorgan funcionalidad a Drupal permitiendo incrementar sus capacidades o adaptarlas a las necesidades de cada portal Web. La siguiente capa de Drupal la constituyen los “*Blocks & Menus*”, estos permiten estructurar y organizar los contenidos en la página Web, es decir que son los elementos que albergan y permiten acceder al usuario a la salida generada y procesada por los módulos a partir de la información. Drupal dispone de un registro de usuarios y de roles que permiten especificar que tareas pueden realizar y a que contenidos puede acceder cada tipo de usuario. Es decir que las operaciones que se pueden realizar sobre los elementos provenientes de las capas inferiores (lectura, modificación, creación) se encuentran limitadas por la capa de control de usuarios y permisos de Drupal.

Capítulo 3. Análisis y Diseño

La última capa, es la capa de “*Templates*” y la cual establece la apariencia gráfica o estilo de la información que se le muestra al usuario. Esta separación entre información y aspecto gráfico permite cambiar el diseño o apariencia del portal Web sin necesidad de modificar los contenidos, lo que es práctico si lo único que queremos es renovar la apariencia. [26]

A continuación se muestra una figura que representa como está estructurada la arquitectura del CMS Drupal:

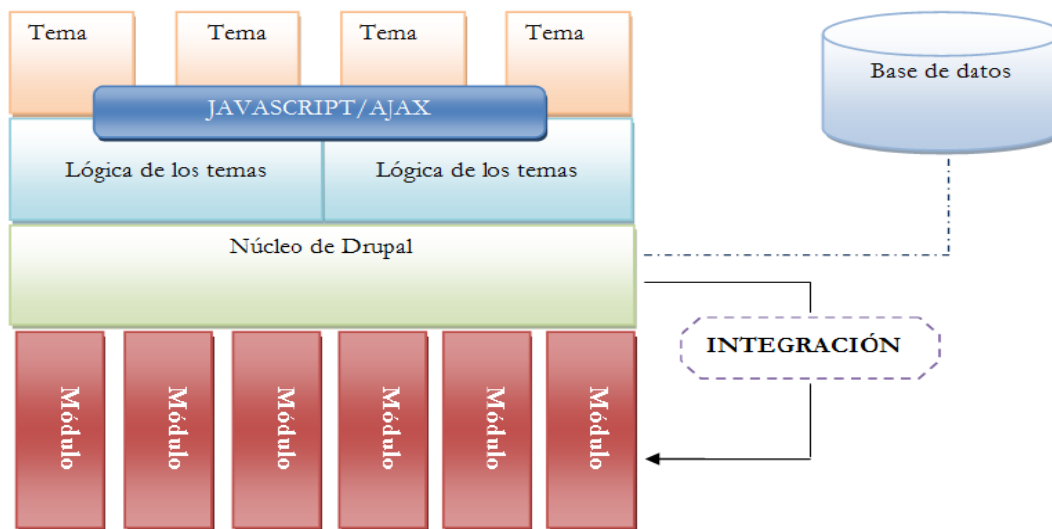


Figura 3. 1: Estructura de Drupal.

Núcleo de Drupal: Formado por un conjunto de librerías que permiten gestionar los procesos de arranque del sistema. Estas librerías ofrecen además un conjunto de servicios que permiten integrar las funcionalidades adicionales de los módulos, servicios como conexión y administración de la base de datos, gestión de procesos de mailing, tratamiento de imágenes, internacionalización, soporte para la codificación y un potente entorno de integración de utilidades. Este último servicio permite ampliar las funcionalidades de un sistema Drupal de una forma relativamente sencilla.

Capítulo 3. Análisis y Diseño

Módulos: Constituyen el núcleo de Drupal y forman parte de la distribución básica del entorno. Tanto los módulos que constituyen el núcleo de Drupal como cualquier otro módulo desarrollado por la comunidad, siguen idénticas directrices en su desarrollo, haciendo uso de las mismas APIs.

Tema: El tema (theme en inglés) define un diseño específico para el sitio web. Existe un repositorio oficial de temas libres de Drupal, que pueden ser descargados y modificados para adaptarlos al diseño de nuestro sitio. Mediante el uso de temas, Drupal separa los contenidos del diseño, de forma que es posible cambiar el aspecto del sitio cambiando o modificando el tema. La forma más fácil de personalizar los temas es modificando las hojas de estilo CSS que contienen.

Base de datos: Esta capa es la encargada de gestionar el acceso a la información almacenada referente al funcionamiento del sistema y a los contenidos que serán mostrados a través del tema activo.

3.1.2 Patrones de Diseño

De los patrones de diseño que presenta Drupal, para el desarrollo del portal se emplean los siguientes:

Singleton (instancia única)

El patrón de diseño Singleton (instancia única) está diseñado para restringir la creación de objetos pertenecientes a una clase o el valor de un tipo a un único objeto. Su intención consiste en garantizar que una clase solo tenga una instancia y proporcionar un punto de acceso global a ella. El patrón Singleton se implementa creando en nuestra clase un método que crea una instancia del objeto sólo si todavía no existe alguna. Las situaciones más habituales de aplicación de este patrón son aquellas en las que dicha clase controla el acceso a un recurso físico único o cuando cierto tipo de datos debe estar disponible para todos los demás objetos de la aplicación. Dentro del core de Drupal se utiliza este patrón de diseño en diversas tareas como la gestión del pool de conexiones con la base de datos y pensando en los módulos y temas de Drupal como objetos para llevar a cabo la gestión de dichos elementos. [27]

Observer (observador)

El patrón Observer define una dependencia del tipo uno-a-muchos entre objetos, de manera que cuando uno de los objetos cambia su estado, el observador se encarga de notificar este cambio a todos los otros dependientes. El objetivo de este patrón es desacoplar la clase de los objetos clientes del objeto,

Capítulo 3. Análisis y Diseño

aumentando la modularidad del lenguaje, así como evitar bucles de actualización (espera activa o polling). Al igual que en el caso del patrón de diseño Decorator este módulo es utilizado para llevar a cabo la extensión de los componentes internos de Drupal a través de los correspondientes *hooks*. [27]

En el módulo “boletines” se implementa el *hook boletines_node_insert()*, que se ejecuta al insertar una entidad de tipo Boletín, por lo que el módulo es declarado como observador ya que se encuentra a la espera de un evento de inserción.

Command (comando)

Este patrón permite solicitar una operación a un objeto sin conocer realmente el contenido de esta operación, ni el receptor real de la misma. Para ello se encapsula la petición como un objeto, con lo que además se facilita la parametrización de los métodos:

- Encapsula un mensaje como un objeto, con lo que permite gestionar colas o registro de mensaje y deshacer operaciones.
- Soportar restaurar el estado a partir de un momento dado.
- Ofrecer una interfaz común que permita invocar las acciones de forma uniforme y extender el sistema con nuevas acciones de forma más sencilla. [27]

Este patrón de diseño es utilizado en Drupal para permitir llevar a cabo la ejecución de ciertas tareas pasando como parámetro el operador, esta es la base fundamental del funcionamiento de los *hooks*.

Chain of Responsibility (cadena de responsabilidades)

El sistema de menús de Drupal sigue este patrón. En cada solicitud de la página, el menú del sistema determina si hay un módulo para gestionar la solicitud y si el usuario tiene acceso a los recursos solicitados. Para ello, el mensaje se pasa a la opción del menú correspondiente a la vía de la solicitud. Si el elemento de menú no puede manejar la petición, se pasa a otro. Esto continúa hasta que un módulo se encarga de la petición, un módulo niega el acceso para el usuario, o la cadena se ha agotado. [27]

En el módulo boletines mediante el *hook boletines_menu()* en dependencia de la petición recibida por el módulo, se delega la responsabilidad a la función encargada de atenderla. De esta forma se continúa la

Capítulo 3. Análisis y Diseño

cadena hasta que un módulo atienda la petición, hasta que un módulo deniegue el acceso o hasta que la cadena se agote. También se evidencia en los menús, en el menú superior y en el de usuarios.

Bridge (puente)

La capa de abstracción de la bases de datos de Drupal es similar al patrón de diseño bridge. Los módulos necesitan ser escritos de forma tal que sean independientes del gestor de bases de datos que se esté usando, y proporciona la capa de abstracción para ello. Permite añadir soporte para otros gestores de bases de datos sin la necesidad de modificar el código del módulo. [27]

Este patrón se evidencia en el módulo boletines que fue programado logrando independencia del motor de BD, que se utilice la función *boletines_node_insert()* que permite conectarse a la BD y realizar consultas abstrayéndose del SGBD que se emplee para manejar los datos almacenados.

Decorator (decorador)

El patrón Decorator responde a la necesidad de añadir dinámicamente funcionalidad a un objeto. Esto nos permite no tener que crear sucesivas clases que hereden de la primera incorporando la nueva funcionalidad, sino otras que la implementan y se asocian a la primera. Este patrón de diseño es utilizado en el core de Drupal para facilitar la extensión de sus componentes. [27]

3.2 Diagramas de Clases de Diseño

En el Diagrama de Clases de Diseño se especifica la estructura de clases del sistema con relaciones entre clases y estructuras de herencia. Durante el análisis del sistema, el diagrama se desarrolla buscando una solución ideal. Durante el diseño, se usa el mismo diagrama, y se modifica para satisfacer los detalles de las implementaciones. [28]

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

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

Capítulo 3. Análisis y Diseño

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

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

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

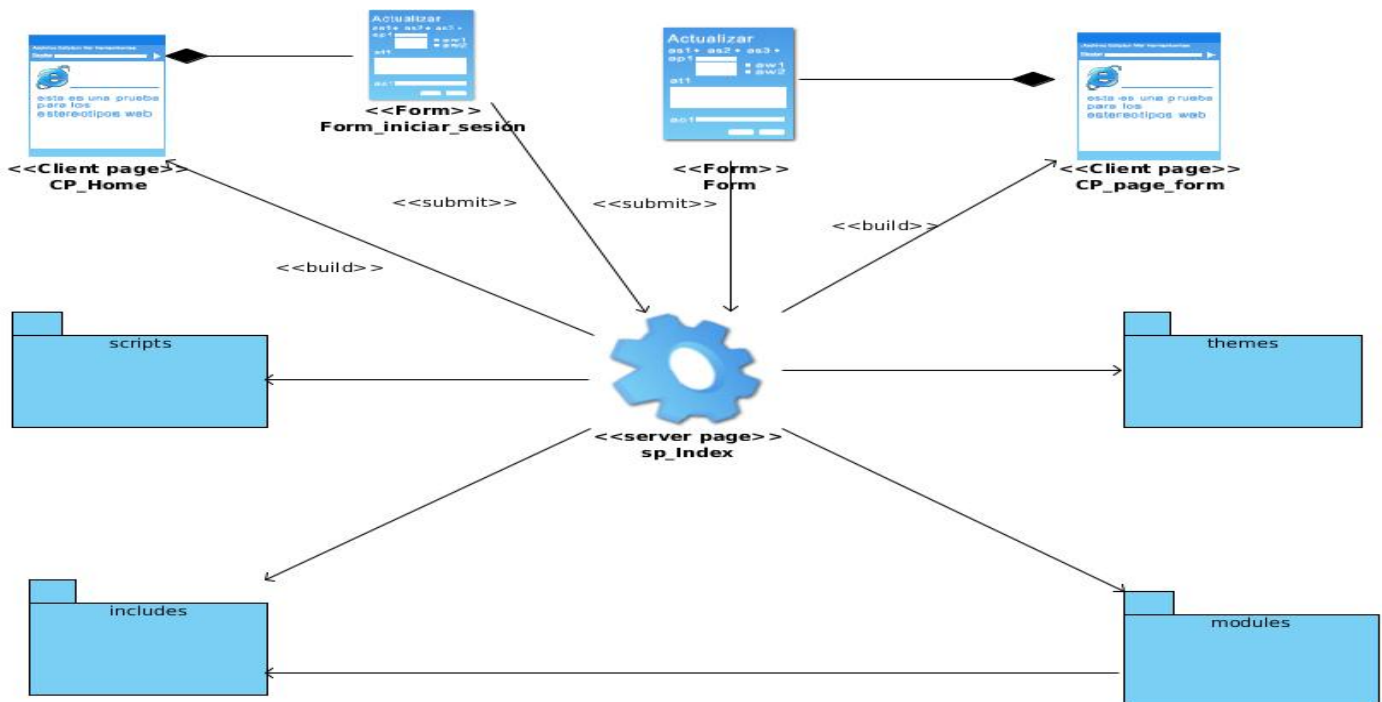


Figura 3. 2: Diagrama de clases de Diseño para Drupal.

A continuación se muestra un diagrama de clases del diseño relacionado con uno de los requisitos funcionales críticos identificados anteriormente.

Diagrama de Clases de Diseño Crear Artículo

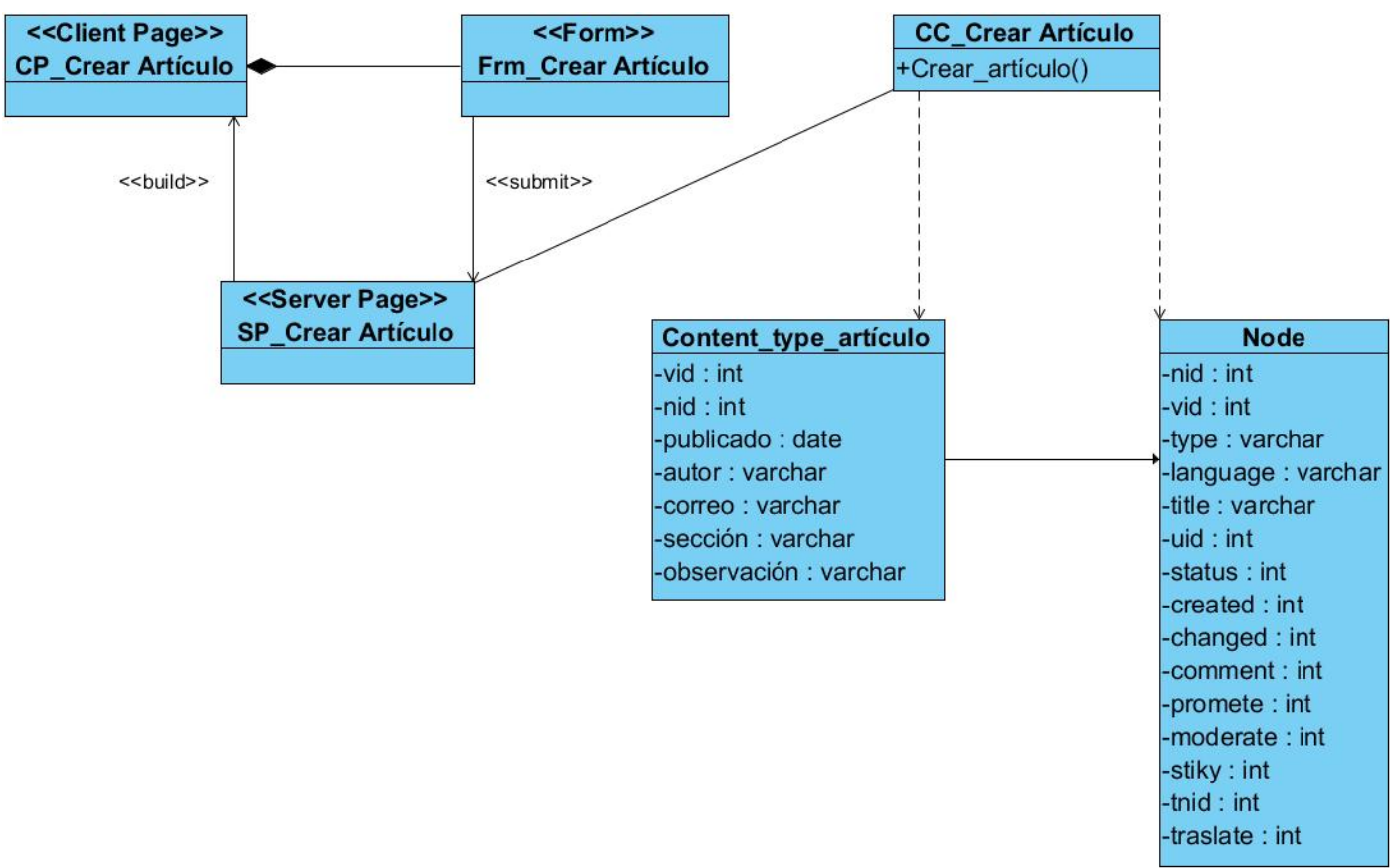


Figura 3. 3: Diagrama de Clases de Diseño Crear Artículo.

En el anexo 1 se evidencian el resto de los diagramas de clases del diseño.

3.3 Diagramas de Secuencia

El diagrama de secuencia de un sistema muestra gráficamente los eventos que originan los actores y que impactan al sistema. La creación de los diagramas de secuencia depende de la formulación de los casos de uso. Durante la operación del sistema, los actores generan eventos, solicitando alguna operación a cambio. [29]

A continuación se muestran dos diagramas de secuencia relacionados con los requisitos funcionales críticos identificados anteriormente.

Diagrama de Secuencia Crear Artículo

Capítulo 3. Análisis y Diseño

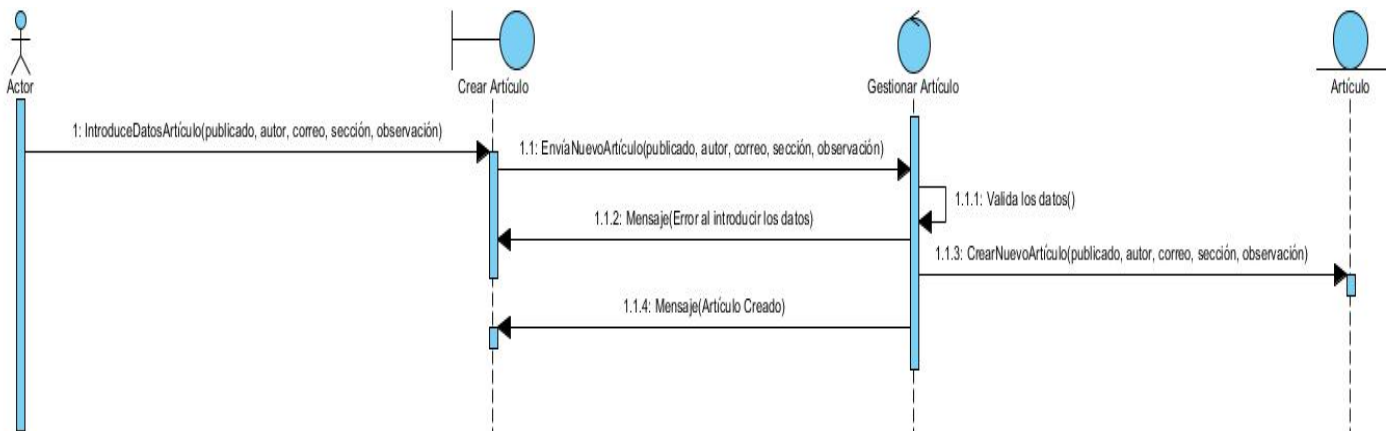


Figura 3. 4: Diagrama de Secuencia Crear Artículo.

Diagrama de Secuencia Crear Concurso

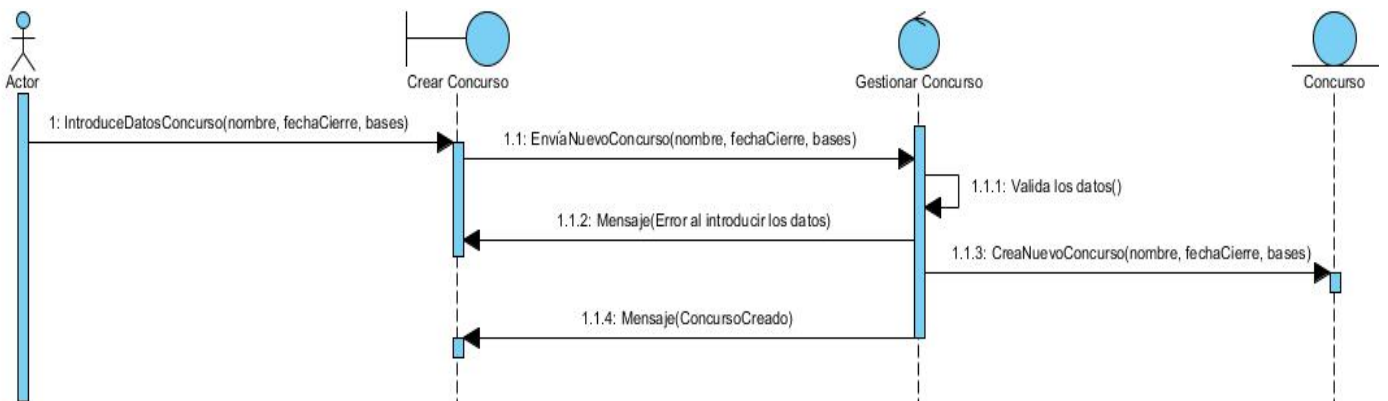


Figura 3. 5: Diagrama de Secuencia Crear Concurso.

En el anexo 2 se evidencian el resto de los diagramas de secuencia.

3.4 Diagrama de Despliegue

El Modelo de Despliegue es un modelo de objetos compuesto por nodos y sus relaciones, que describen la distribución física de un sistema.

El nodo Cliente identifica a cualquier ordenador que a través de la red física usando un navegador se comunicará mediante el protocolo https, con un Servidor Web que le garantizará el acceso al portal Web

Capítulo 3. Análisis y Diseño

de la Revista Pionero donde podrá entrar, informarse y usar los servicios disponibles. Un Servidor de Base de Datos se encargará de mantener las informaciones generadas como parte de la base de datos del CMS Drupal y garantizará el acceso a ella mediante el Servidor Web, ambos servidores se comunicarán mediante el protocolo TCP/IP.

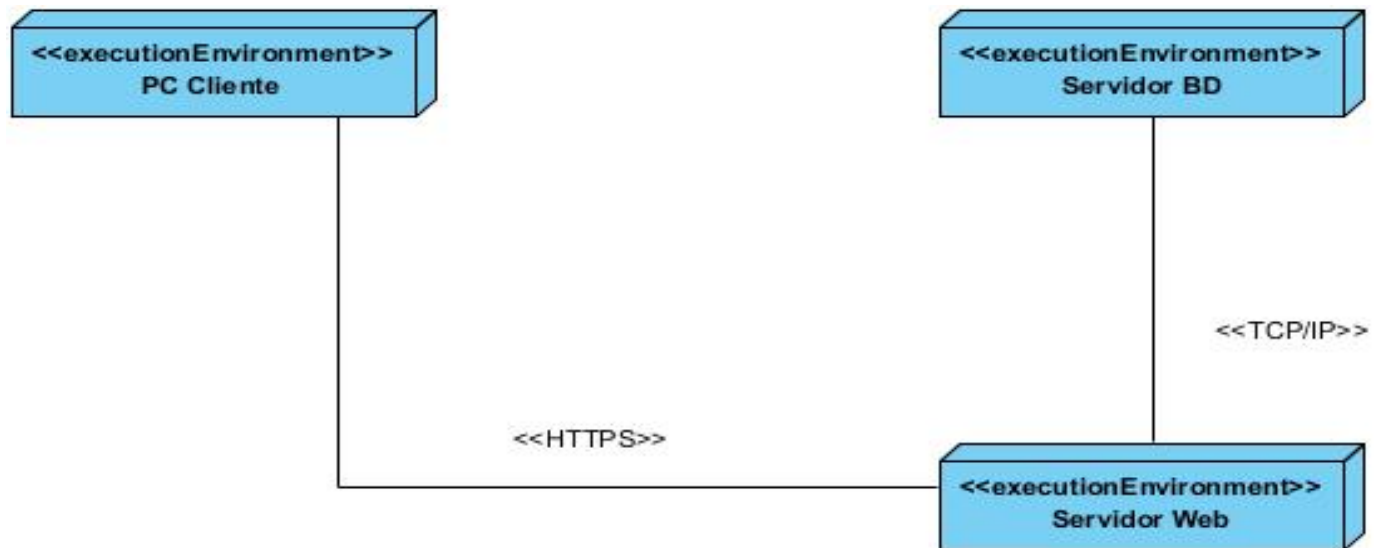


Figura 3. 6: Diagrama de Despliegue.

Conclusiones parciales

Durante el desarrollo de este capítulo se definieron los aspectos relacionados con el análisis y diseño de la aplicación. La selección de Drupal como CMS para el desarrollo del sistema permitió que se heredara su estilo arquitectónico, que es robusto, y constituye un elemento de vital importancia pues la arquitectura define en cuál medida se permitirá o impedirá satisfacer los atributos de calidad del software y juega un papel fundamental para guiar el desarrollo. Con el nivel de comprensión alcanzado, se obtuvo como resultado una representación de la estructura del sistema con el nivel de detalle suficiente para proporcionar una vista general del sistema que diera paso a su posterior implementación.

Capítulo 4. Implementación y Pruebas

Capítulo 4. Implementación y Pruebas

En este capítulo se realizará el desarrollo de la aplicación teniendo en cuenta el modelado que se obtuvo del análisis y diseño planteado en el capítulo anterior, dando cumplimiento a todos los requisitos trazados. Además se efectuarán las pruebas de funcionalidad, carga y estrés, así como las de seguridad para la comprobación del sistema automatizado y las funcionalidades de la revista.

4.1 Diagrama de componentes

Un diagrama de componentes describe la organización de los componentes físicos de un sistema. [30]

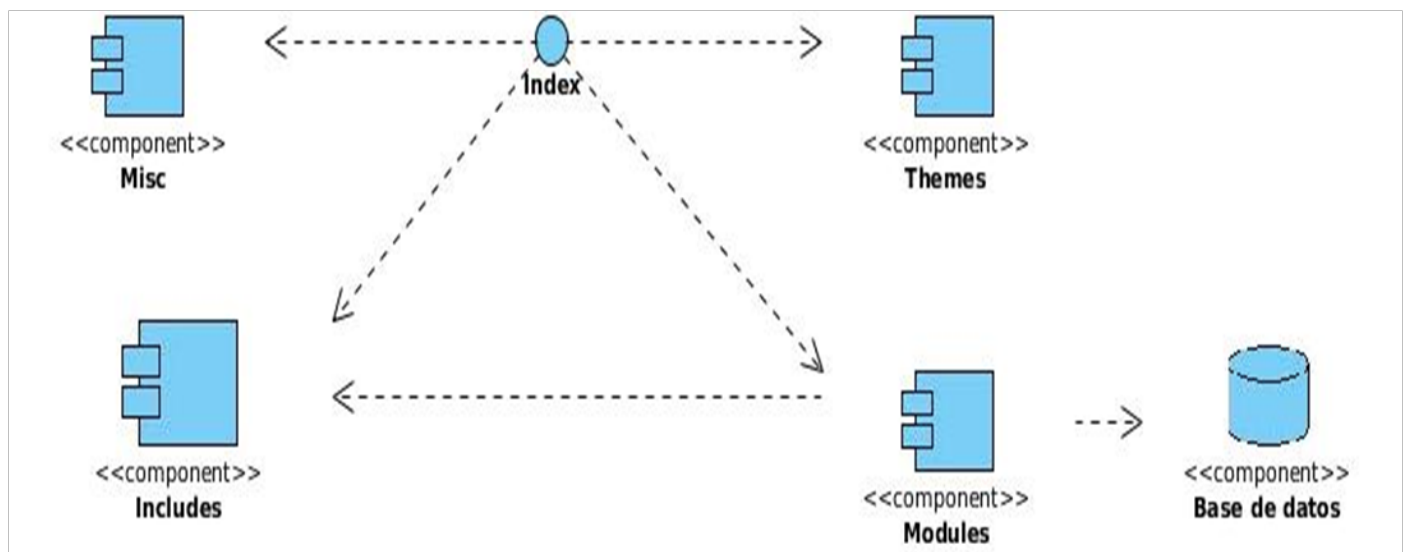


Figura 5. 1: Diagrama de Componentes.

4.2 Diseño de la Base de Datos

El diseño de la base de datos del sistema que se utiliza se hace sencillo con el uso del CMS Drupal. Dicha herramienta tiene una estructura de datos propia que contiene todo lo necesario para que la aplicación que se implementa sea eficiente. A continuación se describe mediante el modelo de datos las principales clases en la base de datos del sistema a desarrollar, así como la forma y descripción de las tablas en sí mismas.

Capítulo 4. Implementación y Pruebas

4.2.1 Modelo de Datos

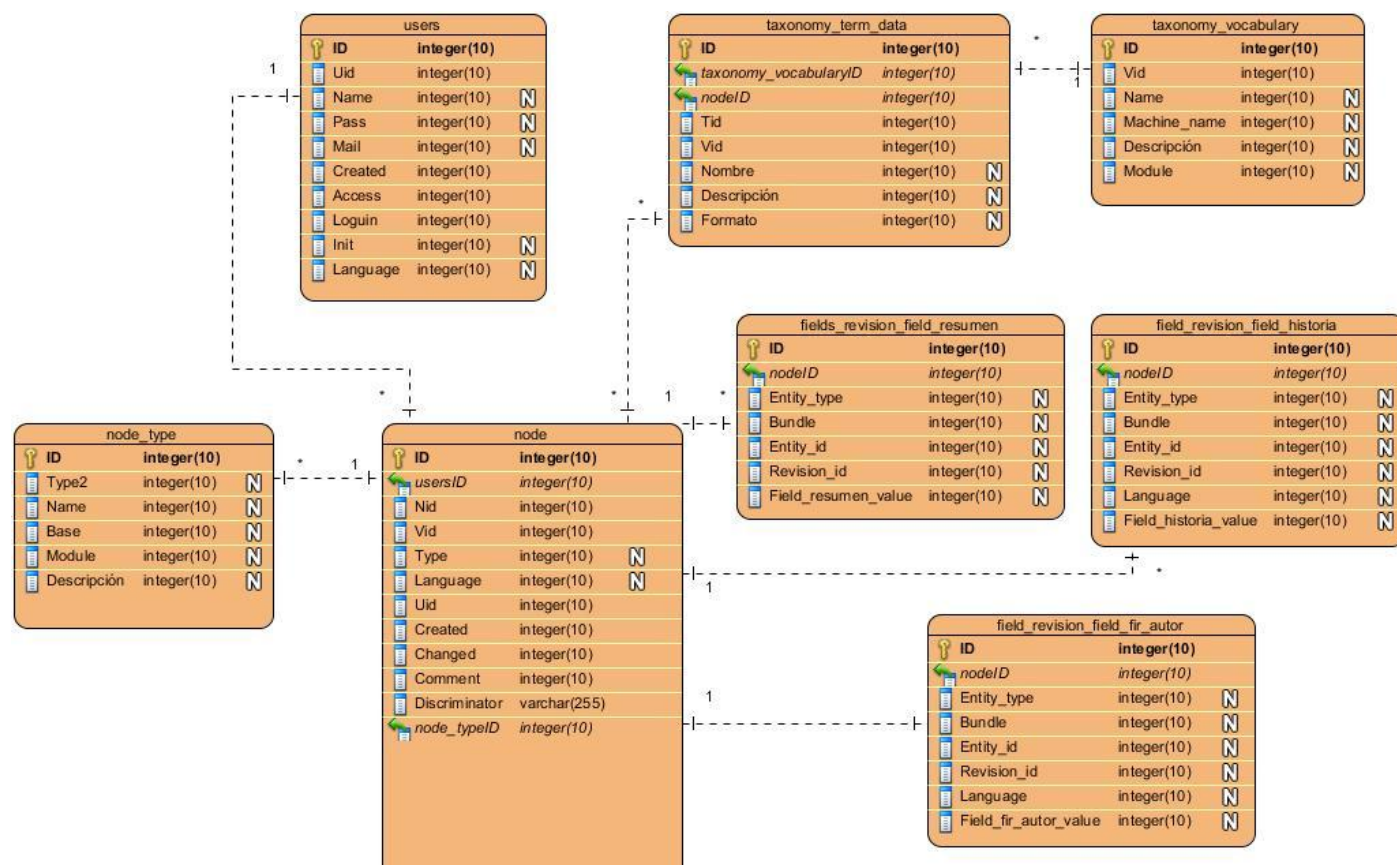


Figura 5. 2: Modelo de Datos

4.3 Código fuente

Estándares de codificación de Drupal

Los estándares de codificación definidos por los desarrolladores de Drupal son:

Indentación: La indentación consiste en insertar espacios en blanco o tabuladores en determinadas líneas de código para facilitar su comprensión. En programación se utiliza la indentación para anidar elementos.

Etiquetas de apertura y cierre de PHP: Cuando estemos escribiendo en PHP, siempre se deben utilizar las etiquetas `<?php` y `?>`, y en ningún caso la versión corta `<?` y `?>`. En general se omite la etiqueta de cierre de PHP (`?>`) al final de los archivos **.module** y **.inc**. Esta convención evita que se puedan quedar

Capítulo 4. Implementación y Pruebas

olvidados espacios no deseados al final del archivo (después de la etiqueta de cierre `?>`), que serían identificados como salida HTML y podrían provocar un error muy típico, *"Cannot modify header information - headers already sent by..."*.

Operadores: Los operadores binarios, que se utilizan entre dos valores, deben separarse de estos valores, a ambos lados del operador, por un espacio. Por ejemplo, **\$numero = 3**, en lugar de `$numero=3`. Esto se aplica a operadores como `+`, `-`, `*`, `/`, `=`, `==`, `!=`, `>`, `<`, `.` (concatenación de cadenas), `.=`, `+=`, `-=`, etc. Los operadores unarios como `++`, `--` no deben tener separación. Por ejemplo, **\$numero++**.

Uso de comillas: Se pueden usar tanto las comillas simples ('cadena') como las comillas dobles ("cadena") para delimitar las cadenas de caracteres. Las comillas dobles son necesarias si se desean incluir variables dentro de las cadenas de texto. Por ejemplo, `<h1>$title</h1>`. También se recomienda el uso de comillas dobles cuando el texto puede incluir alguna comilla simple.

Uso de punto y coma (;) en código PHP: Aunque PHP permite escribir líneas de código individuales sin el terminador de línea (;), como por ejemplo `<?php print $title ?>`. En Drupal es siempre obligatorio: `<?php print $title; ?>`.

Estructuras de control: Con respecto a las estructuras de control, hay que tener en cuenta las siguientes normas:

- Debe haber un espacio entre el comando que define la estructura (`if`, `while`, `for`, etc.) y el paréntesis de apertura. Esto es así para no confundir las estructuras de control con la nomenclatura de las funciones.
- La llave de apertura `{` se situará en la misma línea que la definición de la estructura, separada por un espacio.
- Se recomienda usar siempre las llaves `}` aún en los casos en que no sea obligatorio su uso (una sola "línea" de código dentro de la estructura de control).
- Las estructuras `else` y `elseif` se escribirán en la línea siguiente al cierre de la sentencia anterior.

Arrays: Los valores dentro de un array (o matriz) se deben separar por un espacio (después de la coma que los separa). El operador `=>` debe separarse por un espacio a ambos lados.

Cuando la línea de declaración del array supera los 80 caracteres, cada elemento se debe escribir en una única línea, indentándolo una vez (2 espacios). En este último caso, la coma de separación del último elemento también se escribirá, aunque no existan más elementos.

Variables globales: Aunque el uso de variables globales está desaconsejado, en caso de necesitarse, éstas se declararán utilizando un guión bajo inicial, seguido del nombre del módulo o tema y otro guión bajo antes del nombre de la variable.

Capítulo 4. Implementación y Pruebas

URLs de ejemplo: Por convención, siempre que haya que indicar una URL de ejemplo, se debe utilizar "example.com". Por ejemplo: `http://example.com/node/add/article`.

Idioma: Drupal utiliza el idioma inglés como base para todo el sistema. Es por ello que todos los módulos deberían escribirse en inglés, tanto su código (nombres de funciones, variables, etc.) como las cadenas de texto que se mostrarán al usuario. Para que estas cadenas sean traducibles se utilizará la función de traducción `t()`.

Comentar el código: En este apartado debemos diferenciar entre los comentarios para aclarar determinados fragmentos de código, que se insertan en cualquier punto del mismo, y los comentarios de documentación. Los comentarios de documentación suelen escribirse al principio de un archivo o de cada función y se utilizan para generar documentación de ayuda a través de aplicaciones que extraen la información a partir de las etiquetas empleadas. En el primero de los casos se pueden utilizar las etiquetas `/* */` para comentarios en varias líneas y `//` para comentarios de una única línea. Se deben escribir frases completas, comenzándolas con mayúscula y terminándolas con un punto. En caso de que en el comentario se haga referencia a una constante, ésta deberá escribirse en mayúsculas (por ejemplo TRUE o FALSE).

Documentando funciones

Todas las funciones que pueden ser llamadas desde otros archivos deben estar comentadas (es recomendable comentar todas las funciones, independientemente del ámbito en que vayan a ser utilizadas). El bloque de comentario debe ir justo encima de la declaración de la función, sin dejar líneas en blanco.

4.4 Pantallas principales de la aplicación

En la siguiente figura se muestra la interfaz principal de la revista Pionero, en ella se ve el menú principal, donde están las secciones ¿Quieres leer? donde se puede desplegar y encontrar varios temas de interés para los adolescentes, también están las galerías, club de amigos, pioneros al día, entre otras.

Capítulo 4. Implementación y Pruebas

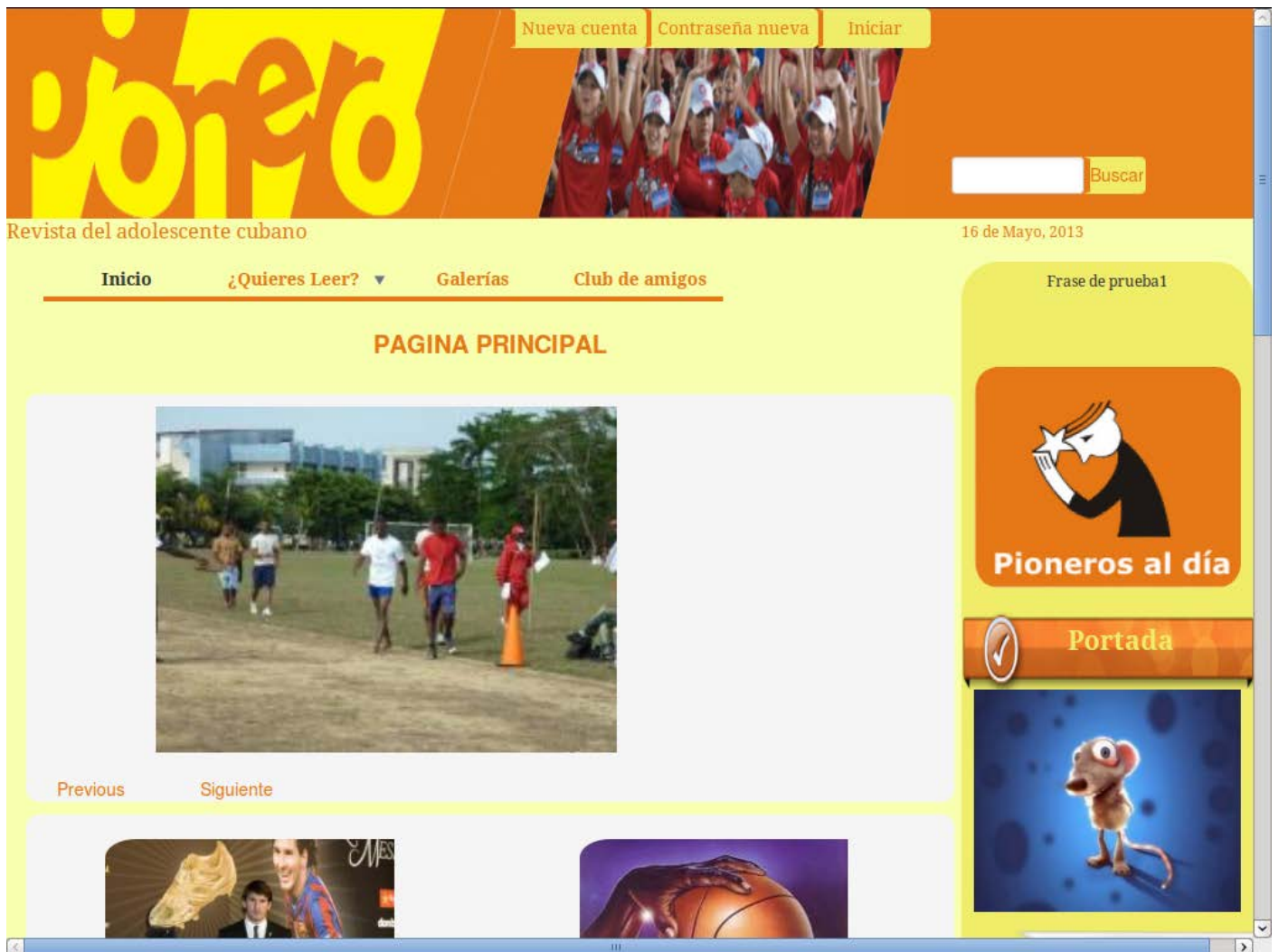


Figura 5. 3: Pantalla principal de la aplicación.

A continuación se muestra la sección Pioneros al Día, donde se mantienen informados a los adolescentes de las actividades que se llevarán a cabo por esa fecha.

Capítulo 4. Implementación y Pruebas



Figura 5. 4: Interfaz de la sección Pioneros al Día.

En la siguiente figura se muestra la sección Nuestra América, donde se muestran artículos relacionados con el continente americano, para que así los pioneros busquen información sobre las curiosidades de América Latina.

Capítulo 4. Implementación y Pruebas

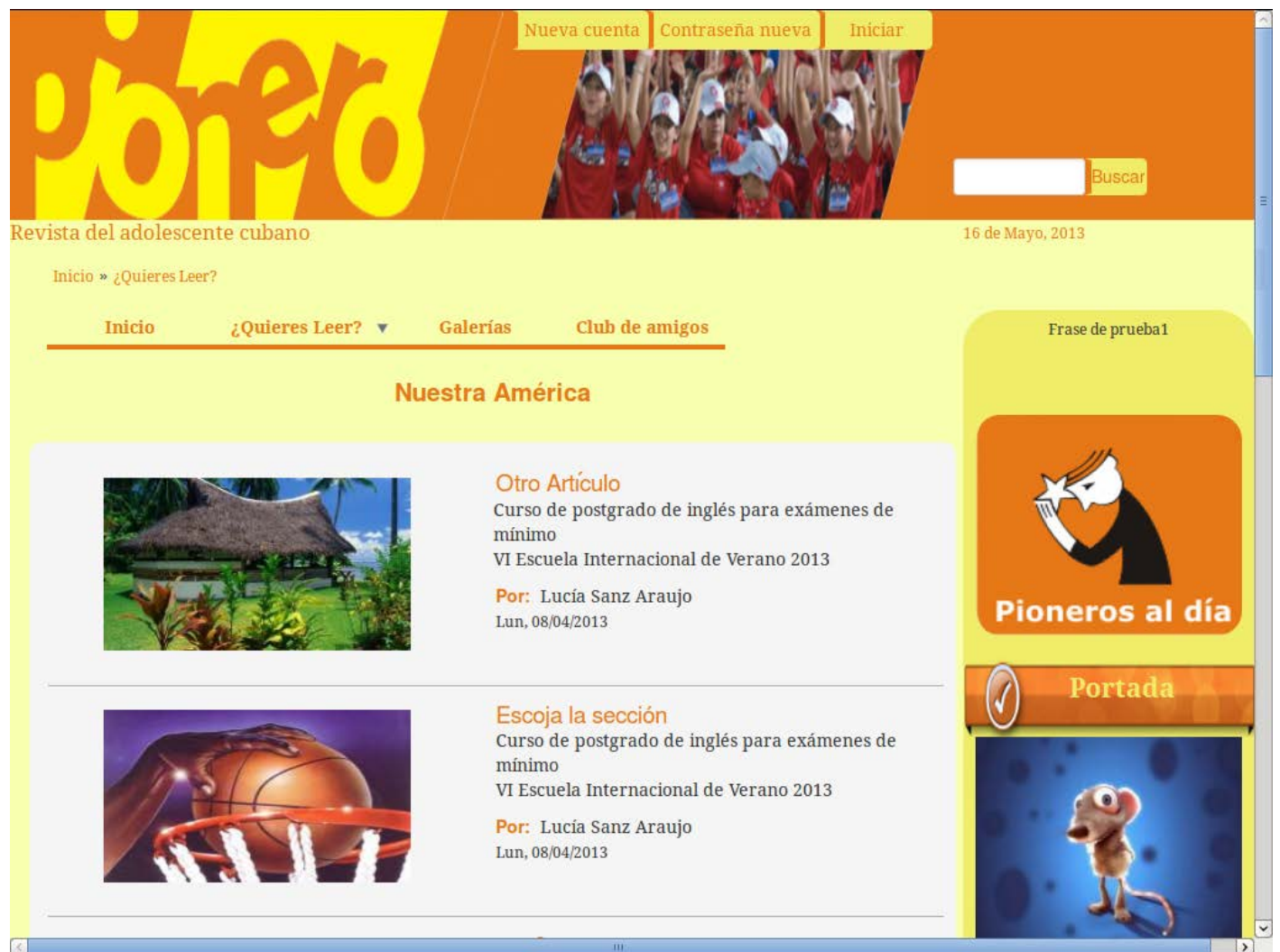


Figura 5. 5: Interfaz de la sección Nuestra América.

4.5 Validación del sistema

En el siguiente acápite se mostrarán los principales resultados de las pruebas realizadas al portal Web.

4.5.1 Pruebas Funcionales

Se diseñaron casos de pruebas para la realización de las pruebas funcionales al sistema, algunos se muestran a continuación.

Capítulo 4. Implementación y Pruebas

Crear Concurso

Condiciones de ejecución

- El usuario debe tener el rol Editor.
- El usuario debe encontrarse autenticado en el sistema.

Descripción de las Variables

En la siguiente tabla se muestra la descripción de cada una de las variables del requisito funcional Crear Concurso.

No	Nombre de campo	Clasificación	Valor Nulo	Descripción
1.	Nombre	Campo de Texto.	No	
2.	Tipo	Lista Desplegable.	No	Se escoge un término de una taxonomía.
3.	Fecha Cierre	Campo de Texto.	No	Muestra menú emergente para Fecha.
4.	Bases	Textarea.	No	

Tabla 4. 1: Descripción de los campos del Requisito Crear Concurso.

En la siguiente tabla se muestran las respuestas que muestra el sistema al introducir los diferentes datos.

Escenario	Descripción	Nombre	Tipo	Fecha Cierre	Bases	Respuesta del sistema	Flujo central
EC 1.1 Datos Correctos	Todos los datos son válidos.	V	V	V	V	El sistema muestra el mensaje: Mi OPJM se ha creado.	1- El usuario accede a Adicionar Concurso e inserta los datos y presiona el botón Guardar. 2- El sistema valida los datos y si están correctos, muestra
		Mi OPJM	Organización	12/06/2013	Texto de las Bases.		

Capítulo 4. Implementación y Pruebas

							mensajes en dependencia de su validez. 2.1- Si algún dato es incorrecto el usuario lo corrige y presiona el botón Guardar nuevamente.
EC 1.2 Algún Dato Inválido.	Existe algún dato inválido.	V	V	I	V	El sistema muestra el mensaje: El valor 2604/2013 no coincide con el formato esperado.	1- El usuario accede a Adicionar Concurso e inserta los datos y presiona el botón Guardar. 2- El sistema valida los datos y si están correctos, muestra mensajes en dependencia de su validez. 2.1- Si algún dato es incorrecto el usuario lo corrige y presiona el botón Guardar nuevamente.
		Mi OPJM	Organización	2604/2013	Texto de las Bases.		

Tabla 4. 2: Respuestas del sistema al introducir los datos en los campos del requisito Crear Concurso.

Valoración de las pruebas funcionales realizadas

Se realizaron 3 iteraciones de pruebas funcionales donde se detectaron No Conformidades de acuerdo a las siguientes gráficas.

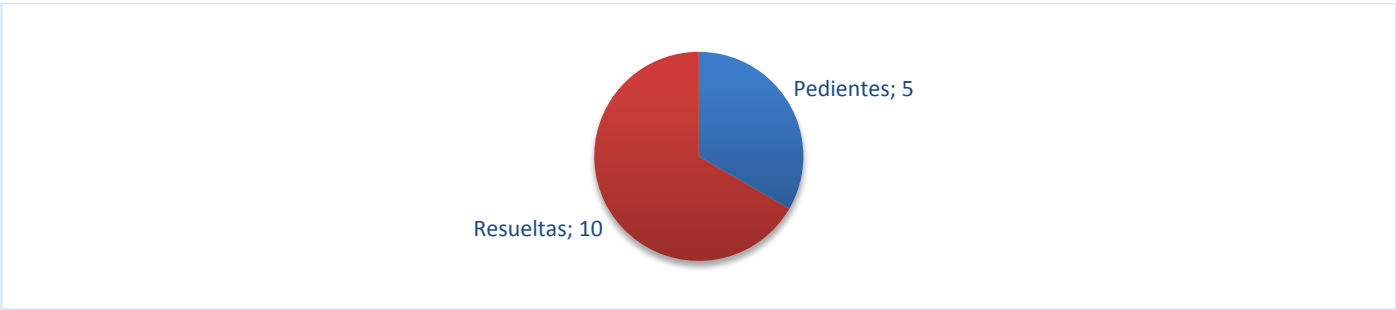


Figura 5. 6: Gráfica de resultados de la iteración 1 de las pruebas funcionales.

En la segunda iteración se detectaron 7 No Conformidades que fueron resueltas junto a las pendientes de la primera iteración.

Capítulo 4. Implementación y Pruebas

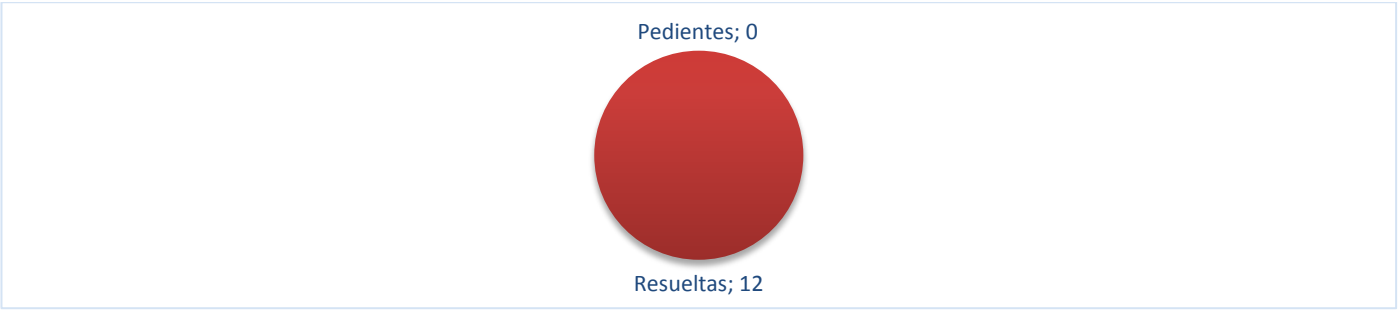


Figura 5. 7: Gráfica de resultados de la iteración 2 de las pruebas funcionales.

En la tercera iteración no se detectaron No conformidades.

La realización de estas pruebas permitió detectar las No Conformidades que presentaba la aplicación, para esto se efectuaron 3 iteraciones logrando la corrección de las mismas en su totalidad.

En el anexo 4 se evidencian el resto de las pruebas funcionales realizadas a la aplicación.

4.5.2 Pruebas de carga y estrés de la aplicación

En la siguiente tabla se muestra las pruebas de carga y estrés realizadas a la aplicación por la herramienta Jmeter para comprobar el rendimiento del sistema.

ID del escenario	Escenarios de la sección	Carga de trabajo	Descripción	Resultado esperado	Resultado de la prueba	Error
EC1- Acceder a la interfaz inicial del portal pioneros.	EC1.1-Acceder a la interfaz inicial del portal pioneros. http://8deabril/Pionero2	30	Esta interfaz es el inicio por tanto es la primera página por la que accederán los usuarios.	La aplicación a respondido en una velocidad menor "50 seg".	14.6 seg.	0.05%

Capítulo 4. Implementación y Pruebas

EC2-Leer.	EC2.1-Mostrar historia Para ello ejecutar click en la pestaña "Historia" perteneciente a "Leer".	30	Esta interfaz muestra información sobre nuestra historia.	La aplicación a respondido en una velocidad menor "50 seg".	14.6 seg.	0.05%
Muestras	Media	Median a	Min	Max	Error	Rendimie nto
1530	30183	2	1	2084079	0.05%	14.6/ seg

Tabla 4. 3: Tabla de las pruebas de carga y estrés realizadas a la aplicación.

La realización de las pruebas de carga y estrés para un total de 30 usuarios concurrentes en 1530 muestras con una media de 30183 dio como resultado un rendimiento de 14.6 usuarios por segundo, siendo un resultado satisfactorio de acuerdo a las características del entorno donde fueron realizadas: 1GB de memoria RAM, 120 GB de disco duro y un procesador Intel® Celeron® de 1,90 GHz por lo que los resultados no son los más exactos posibles.

4.5.3 Pruebas de Seguridad

Las pruebas de seguridad buscan medir la Confidencialidad, Integridad y Disponibilidad de los datos, desde la perspectiva del aplicativo, es decir partiendo a identificar amenazas y riesgos desde el uso o interfaz de usuario final. [31]

Para realizar las pruebas de seguridad se tuvieron en cuenta los roles definidos para interactuar con el portal Web, Editor y Lector. A cada usuario se le asignó un rol de los antes mencionados, con el objetivo de comprobar si estos podían acceder a las funcionalidades del portal sin los privilegios requeridos. Con

Capítulo 4. Implementación y Pruebas

este proceso se comprobó que el usuario con permisos de Editor puede llevar a cabo el proceso de gestionar secciones, artículos, concursos, galerías de imágenes, cartas, enlaces, en fin todos los tipos de contenidos que se desarrollaron en la revista. En tanto con los permisos de un usuario Lector se pueden realizar solamente la búsqueda de contenidos publicados en la revista, se puede además realizar comentarios, es decir que solamente tiene permiso para navegar por el portal pero no puede realizar ninguna modificación en el mismo. En una primera iteración se detectaron errores como: el sistema no protege el envío de datos mediante el protocolo seguro (https), un usuario estándar no puede modificar sus privilegios ni los de otros en la aplicación. Estos errores fueron arreglados correctamente en una segunda iteración, por lo que no hubo que realizar una tercera. Se comprobó además que la aplicación permite el cambio de contraseña únicamente a usuarios autenticados.

Conclusiones parciales

Las pruebas funcionales, de carga y estrés, y de seguridad que se le realizaron a la aplicación evidenciaron la efectividad y la solidez de esta por los resultados satisfactorios obtenidos, consiguiéndose un producto con el menor número de errores posibles, logrando así que la implementación del portal Web obtuviera el resultado esperado por los desarrolladores.

Conclusiones

Conclusiones

Una vez realizada la implementación del portal Web de la revista Pionero, se puede concluir lo siguiente:

- Se caracterizaron las tendencias actuales en el desarrollo de los portales Web dedicados a la difusión de revistas.
- Se diseñó el portal Web de la revista pionero teniéndose en cuenta las funcionalidades con las que debía contar.
- Se implementaron las funcionalidades y servicios definidos para el portal Web de la revista Pionero.
- Se validaron las funcionalidades del portal Web de la revista Pionero.

Dando cumplimiento con los objetivos trazados en esta investigación. Durante la cual se logró publicar el portal Web de la revista Pionero, para que los niños y adolescentes cubanos tuvieran un espacio que visitar y poder acceder a diferentes fuentes de información.

Recomendaciones

Recomendaciones

Los autores del presente trabajo recomiendan la vinculación de la revista con otros medios de divulgación de carácter editorial, lograr la compatibilidad con la mayor cantidad de navegadores posibles, perfeccionar la apariencia del portal Web de la revista a partir de criterios y opiniones obtenidas de los usuarios que la visiten e incluir funcionalidades que pueden ser útiles a los usuarios dependiendo de las tecnologías que se emplean y de los objetivos que puedan ser trazados.

Referencias Bibliográficas

Referencias Bibliográficas

1. Ministerio de la Informática y las Comunicaciones de Cuba (MIC). **¿Qué es INFOSOC?, 2013.**
[Disponible en: <http://www.mic.gov.cu/sitiomic/servlet/hinfosoc>]
2. Las tecnologías de la información y la comunicación (TIC) en la educación: retos y posibilidades.
[Disponible en: <http://www.fundacionsantillana.com>]
3. Revista "Pionero", a cinco décadas de fundada. [Disponible en:
<http://www.radiometropolitana.cu>]
4. Pionero – Definición – WorldReference. [Disponible en:
<http://www.wordreference.com/definicion/pionero>]
5. **Milenium. Principales definiciones de los términos más usados en Internet.** En: Sitio en Internet *Milenium*. [Disponible en:
<http://www.informaticamilenium.com.mx/Paginas/espanol/sitioweb.htm>]
6. **Qué es un CMS** [Disponible en: <http://www.desarrolloweb.com/articulos/que-es-un-cms.html>].
7. [En línea] [Citado el: 21 de Noviembre de 2012.] <http://www.joomlaos.net/caracteristicas-de-joomla>.
8. Castillo, Gleibis Camejo. **Portal de aplicaciones educativas del sistema de educación universitaria en Venezuela. Universidad de Ciencias Informáticas. 2011. Trabajo de Diploma.**
9. **Dries Buytaert.** Drupal Hispano. [En línea] 11 de Abril de 2005. [Citado el: 1 de Diciembre de 2012.] <http://drupal.org.es/caracteristicas>.
10. **Jacobson, I.; Booch, G. y Rumbaugh, J.** *El Proceso Unificado de Desarrollo de Software.* Addison Wesley, 2000.
11. **Eguíluz Pérez, J.** *Introducción a CSS.* 2008.

Referencias Bibliográficas

12. **El lenguaje de programación PHP** [Disponible en: <http://programandoideas.com/el-lenguaje-de-programacion-php/>].
13. **Brandendaugh, J.** *Aplicaciones JavaScript*. Ediciones ANAYA, 2000. [En línea] <http://bibliodoc.uci.cu/pdf/8441510709.pdf>.
14. **Departamento de Informática (ATC, CCIA, LSI)**. Universidad de Valladolid. [Citado el: 15 de 03 de 2010.] [En línea] <http://www.infor.uva.es/~jmrr/tgp/tgprecursos/intro1.htm>.
15. **NetBeans**. *NetBeans*. [En línea] 2010. [Citado el: 10 de Diciembre de 2012.] <http://netbeans.org/>.
16. **Definición Apache (Servidor HTTP) Enciclopedia Proyecto AjpdSoft** [Disponible en: <http://www.ajpdsoft.com/modules.php?name=Encyclopedia&op=content&tid=820>].
17. **¿Qué es un Sistema Gestor de Bases de Datos o SGBD?** [Disponible en: <http://www.cavsi.com/preguntasrespuestas/que-es-un-sistema-gestor-de-bases-de-datos-o-sgbd/>]
18. **Graffito**. [En línea] [Citado el: 1 de Diciembre de 2012.] http://www.graffito.cl/index.php?option=com_content&view=article&id=94:iques-es-qmysqlq&catid=41:preguntas-frecuentes&Itemid=71
19. **Cruz, Yaimara Fernández De la**. *Portal Institucional de Venezolana de Televisión*. Universidad de Ciencias Informáticas. Habana : s.n., 2011. Trabajo de Diploma.
20. **Guatewireless**. [En línea] [Citado el: 1 de Diciembre de 2012.] <http://www.guatewireless.org/articulos/mysql-vs-postgresql.html>.
21. Jordana, Garcilaso. **Introducción a las Metodologías Ágiles**.
22. **Visual Paradigm**. [Citado el: 26 de 2 de 2010.] [En línea] <http://www.visual-paradigm.com/product/vpuml/>.
23. **Modelo de Dominio | Tecnología y Synergix** [Disponible en: <http://synergix.wordpress.com/2008/07/10/modelo-de-dominio/>].

Referencias Bibliográficas

24. Rosa, Rosales Eugenio. **Módulo para la administración de NAS en Nova para servidores. Universidad de Ciencias Informáticas. 2012. Trabajo de Diploma.**
25. **Software y Aplicaciones Web.** [En línea] [Citado el: 21 de Febrero de 2013.]
<http://jtentor.com.ar/post/Arquitectura-de-N-Capas-y-N-Niveles.aspx>.
26. **Tutorial Drupal.** [En línea] [Citado el: 21 de Febrero de 2013.]
<http://www.cursosdrupal.com/content/arquitectura>.
27. [En línea] [Citado el: 22 de Febrero de 2013.] <http://jitcode.blogspot.com/2010/06/poo-y-patrones-de-diseno-en-drupal.html>.
28. **ibiblio. ibiblio.** [En línea] [Citado el: 25 de febrero de 2013.]
<http://www.ibiblio.org/pub/linux/docs/LuCaS/Tutoriales/doc-modelado-sistemas-UML/multiple-html/x219.html>.
29. **Diagrama De Secuencia www.giiaa.com** [Disponible en: <http://www.slideshare.net/giiaa/diagrama-de-secuencia-wwwgiiaacom-presentation>]
30. **diagramas_del_uml** [En línea]
http://www.clubdelsuran.com.ar/site/materiales/proyecto/diagramas_del_uml.pdf.
31. **pruebas de seguridad** [Disponible en: <http://www.vyvquality.com/w1/index.php/servicios/pruebas-de-seguridad.html>]

Anexos

5.1 Anexos 1

Diagrama de Clases de Diseño Editar Artículo

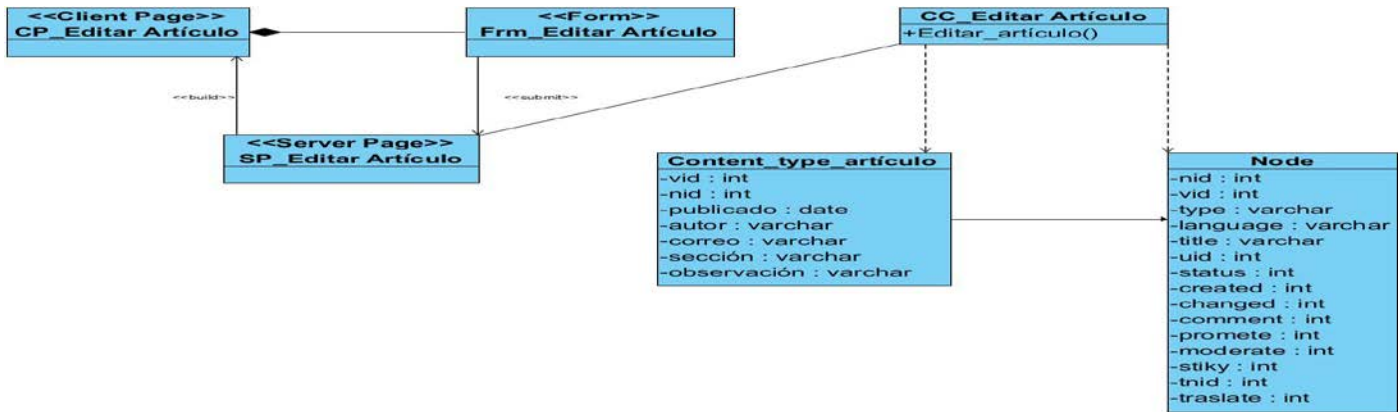


Figura 5. 1: Diagrama de clases del Diseño Editar Artículo

Diagrama de Clases de Diseño Eliminar Artículo

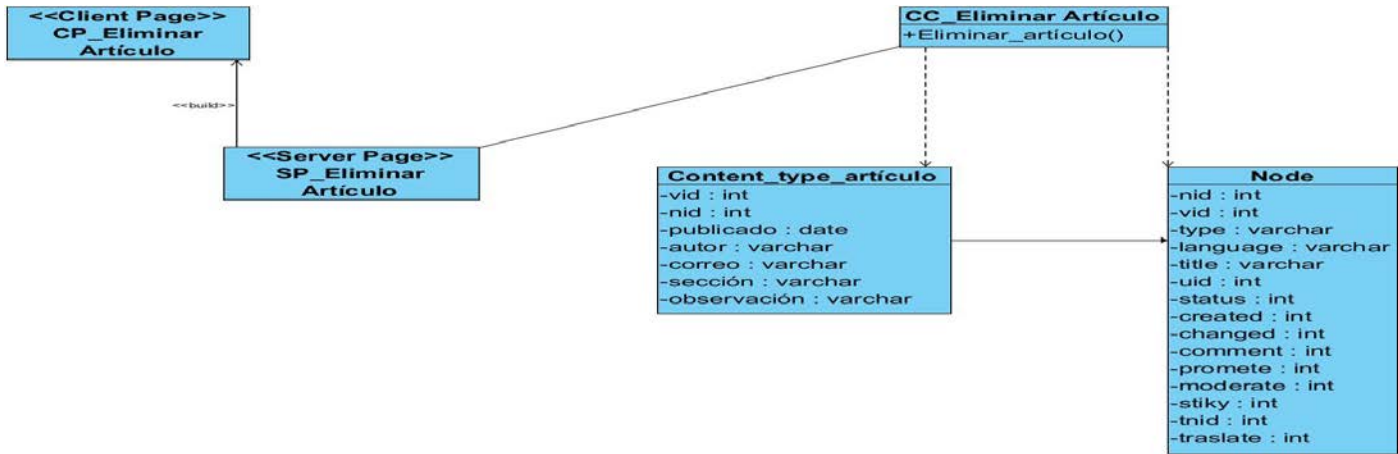


Figura 5. 2: Diagrama de clases del Diseño Eliminar Artículo

Diagrama de Clases de Diseño Listar Artículo

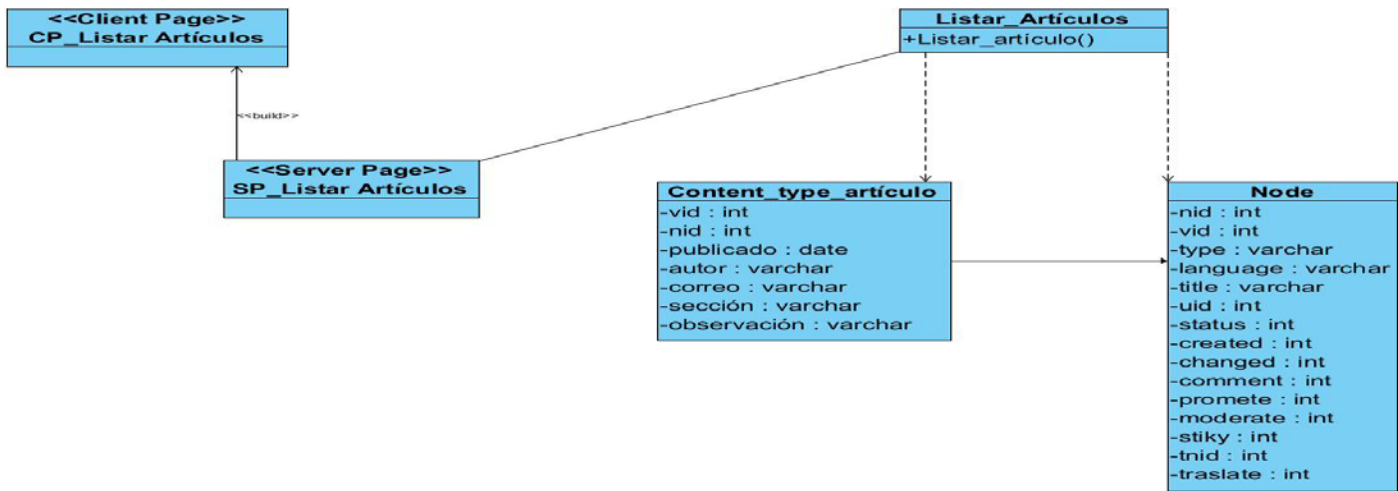


Figura 5. 3: Diagrama de clases del Diseño Listar Artículo

Diagrama de Clases de Diseño Crear Concurso

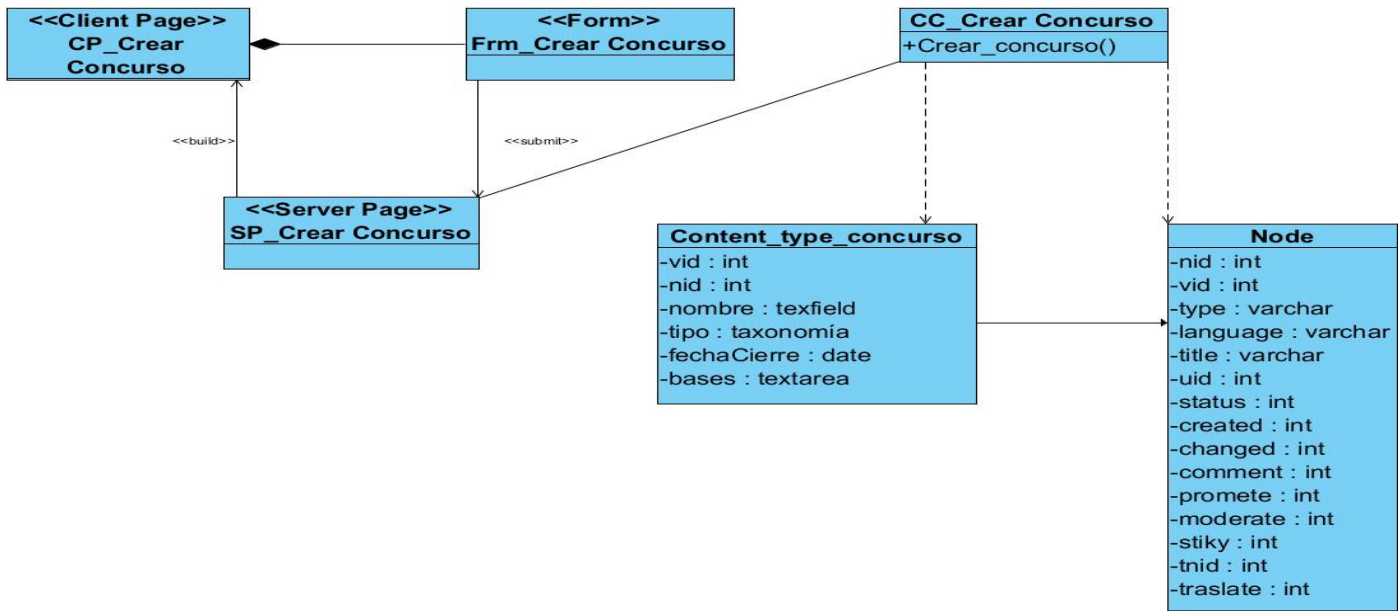


Figura 5. 4: Diagrama de clases del Diseño Crear Concurso

Diagrama de Clases de Diseño Editar Concurso

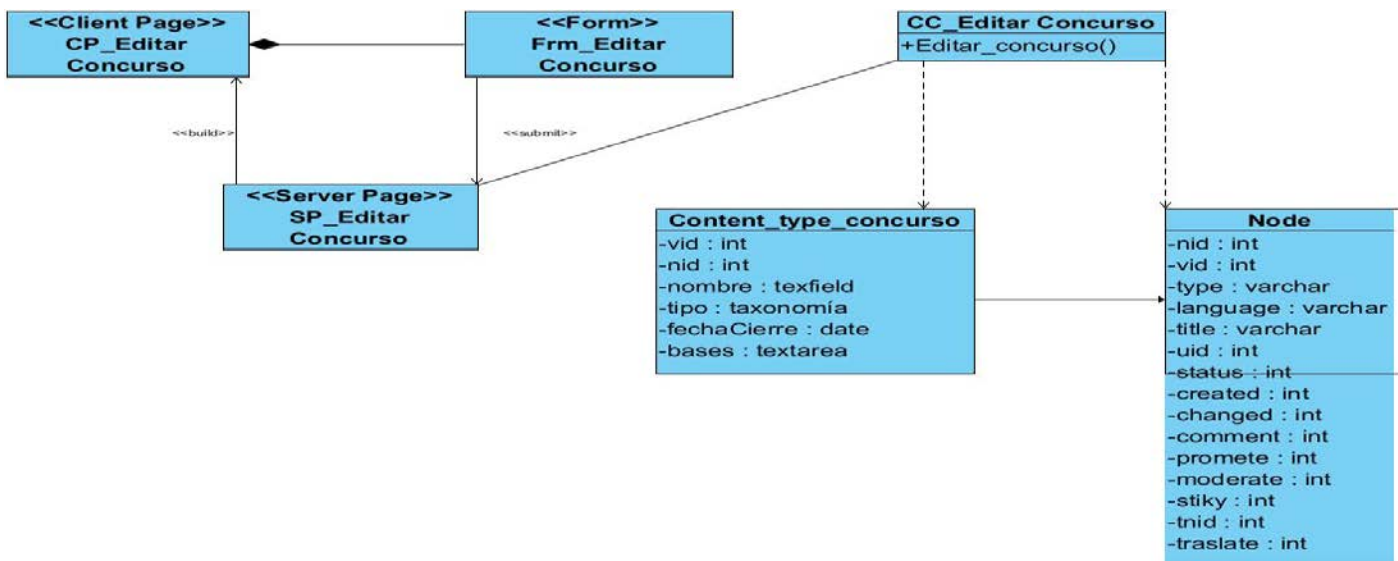


Figura 5. 5: Diagrama de clases del Diseño Editar Concurso

Diagrama de Clases de Diseño Eliminar Concurso

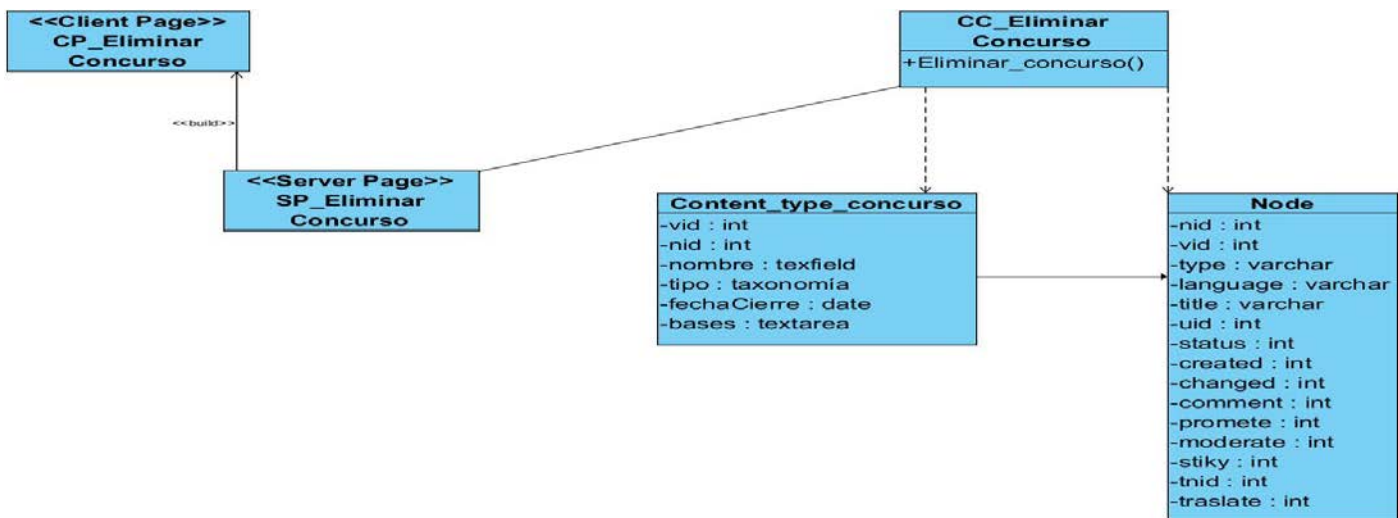


Figura 5. 6: Diagrama de clases del Diseño Eliminar Concurso

Diagrama de Clases de Diseño Listar Concurso

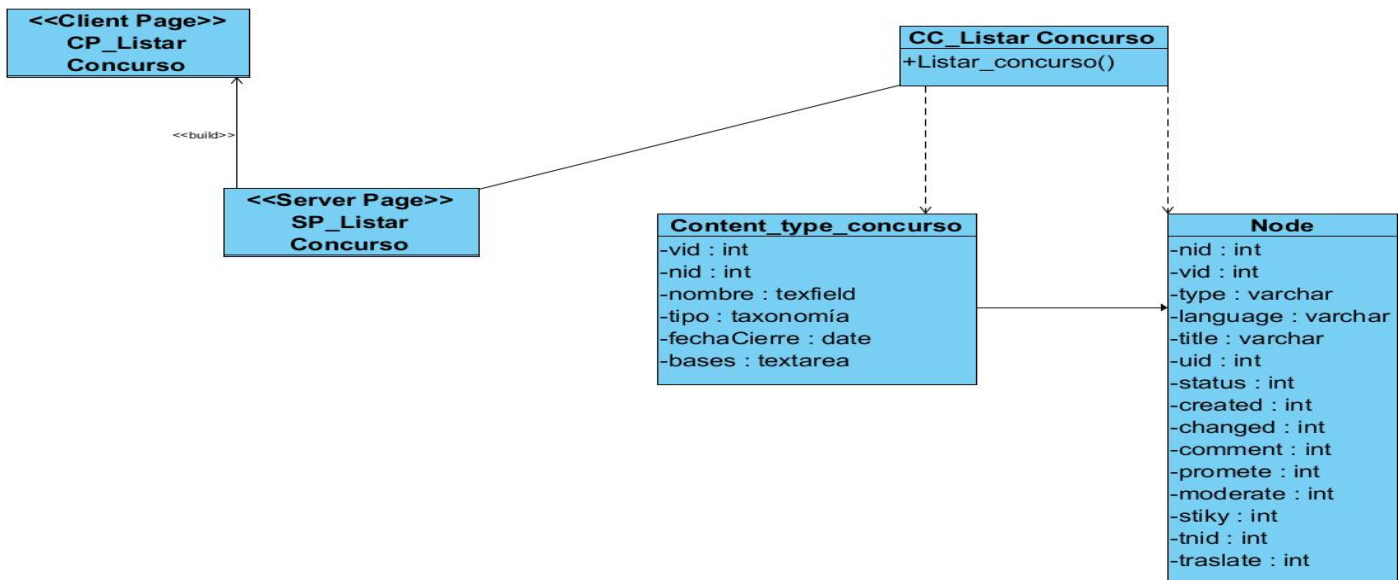


Figura 5. 7: Diagrama de clases del Diseño Listar Concurso

Diagrama de Clases de Diseño Solicitar Inscripción

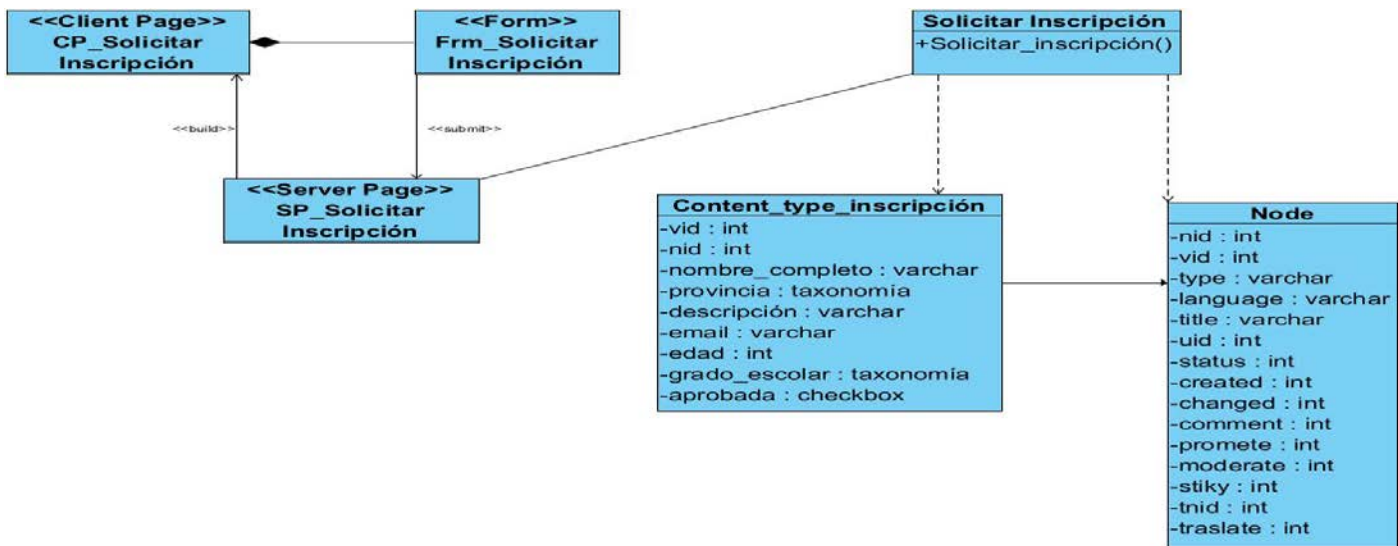


Figura 5. 8: Diagrama de clases del Diseño Solicitar Inscripción

Diagrama de Clases de Diseño Editar Inscripción

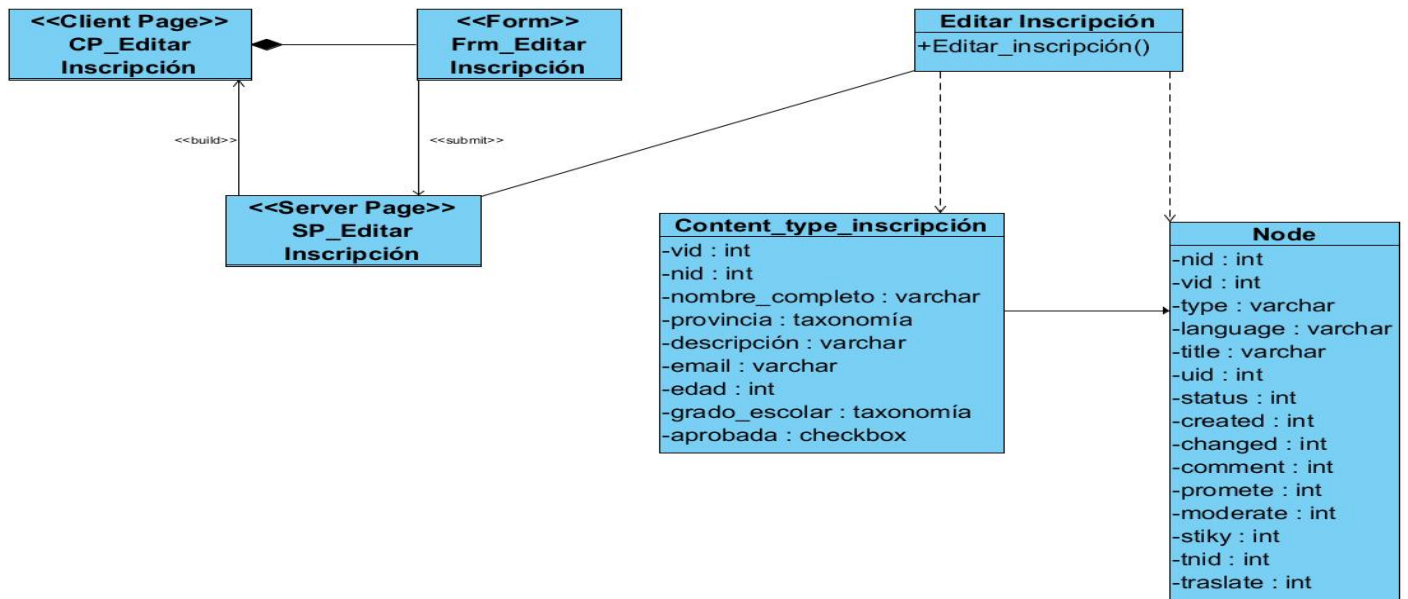


Figura 5. 9: Diagrama de clases del Diseño Editar Inscripción

Diagrama de Clases de Diseño Eliminar Inscripción

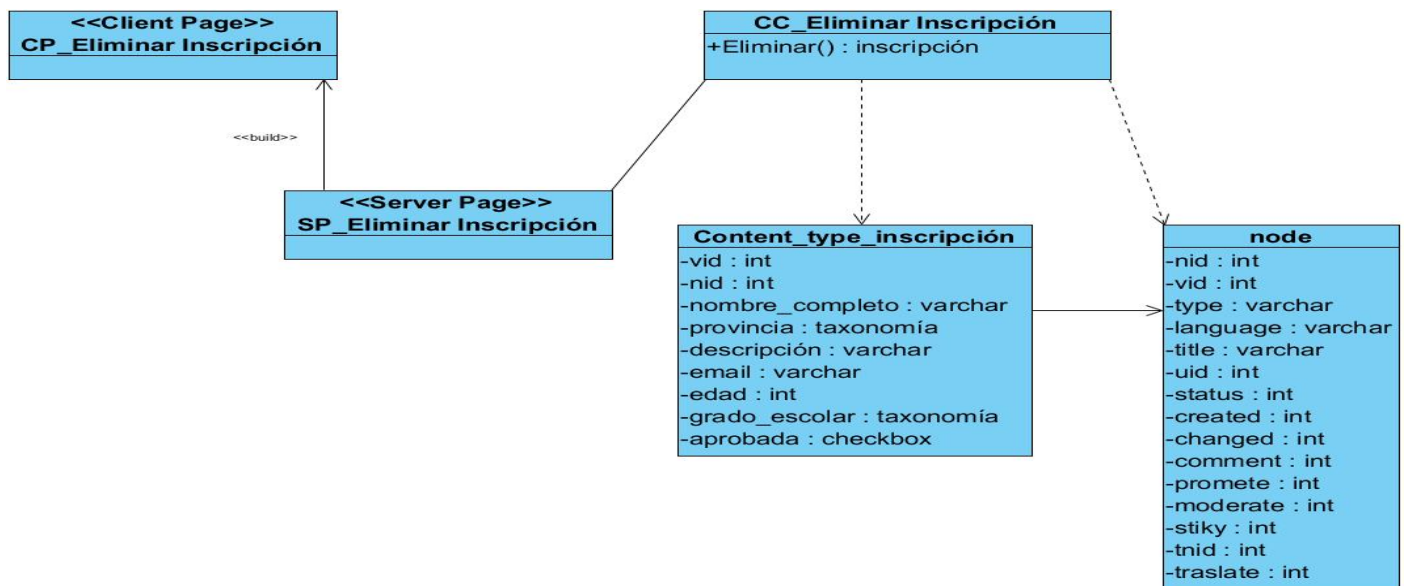


Figura 5. 10: Diagrama de clases del Diseño Eliminar Inscripción

Diagrama de Clases de Diseño Listar Inscripción

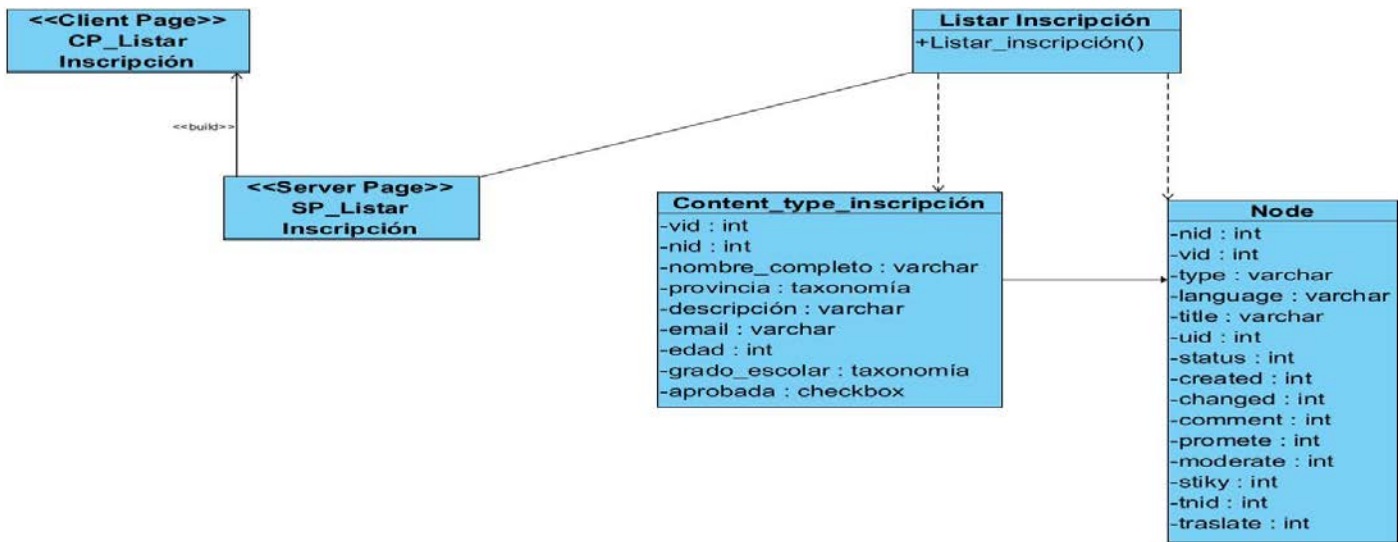


Figura 5. 11: Diagrama de clases del Diseño Listar Inscripción

Diagrama de Clases de Diseño Asignar Rol Usuario

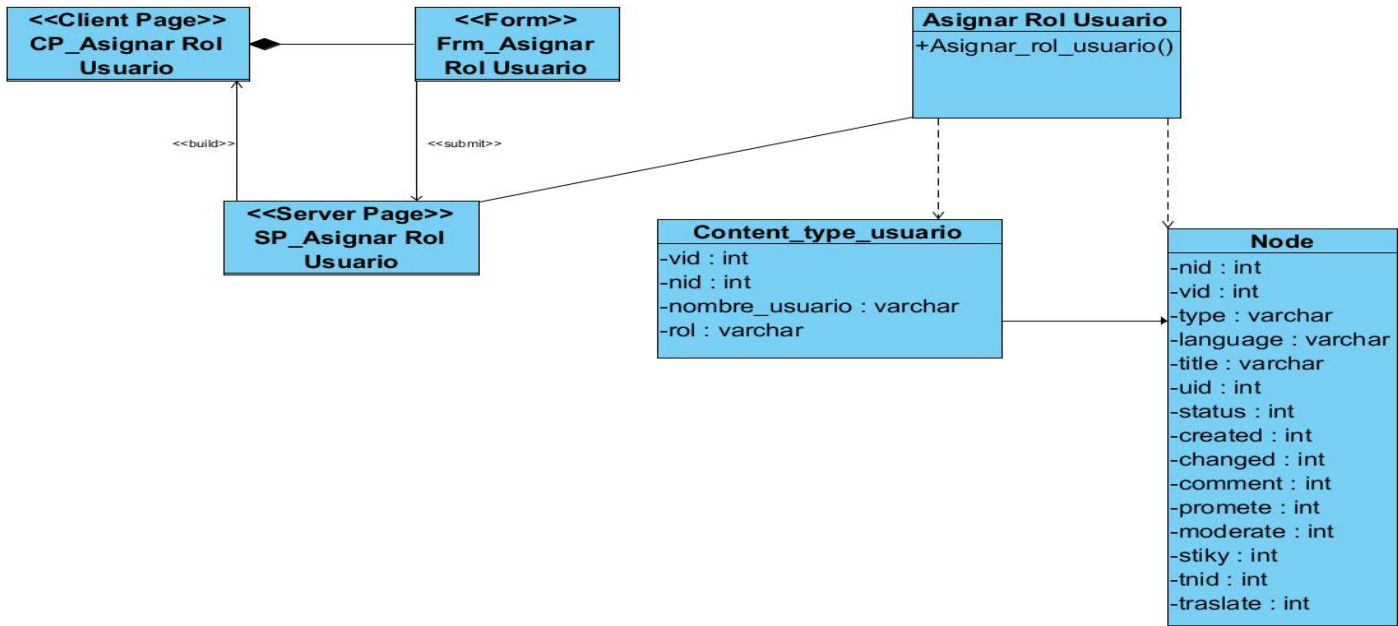


Figura 5. 12: Diagrama de clases del Diseño Asignar Rol Usuario

Diagrama de Clases de Diseño Autenticar Usuario

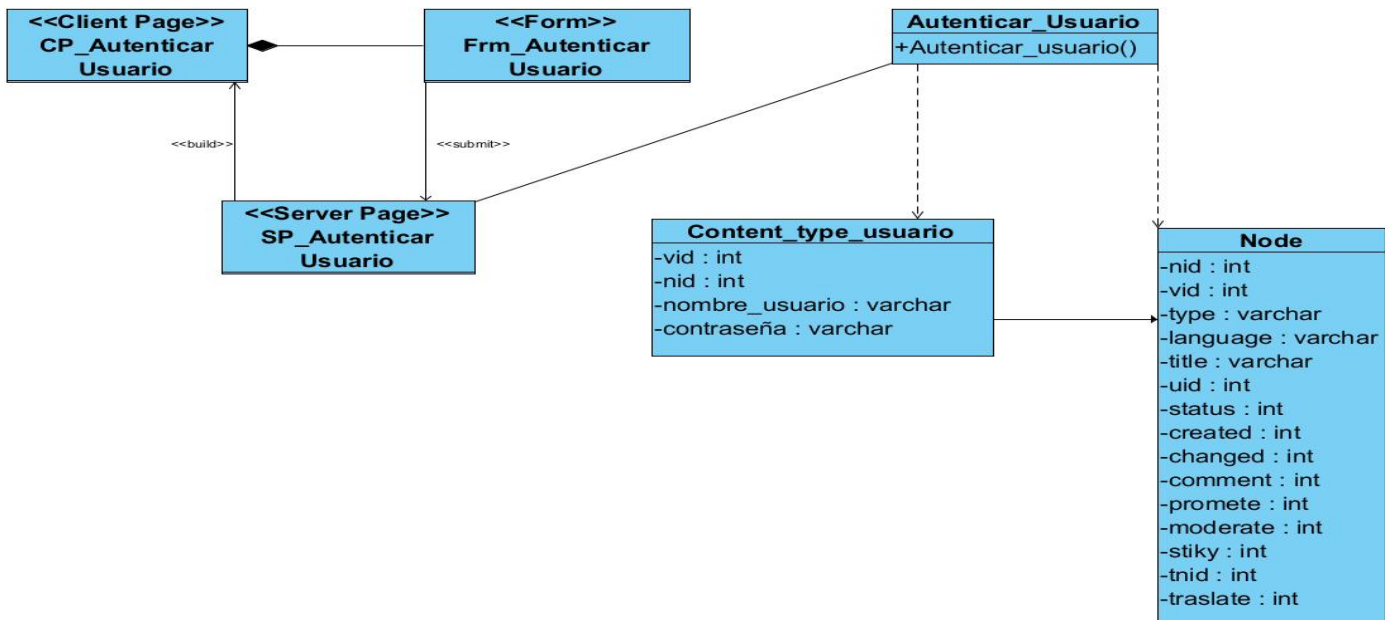


Figura 5. 13: Diagrama de clases del Diseño Autenticar Usuario

Diagrama de Clases de Diseño Crear Usuario

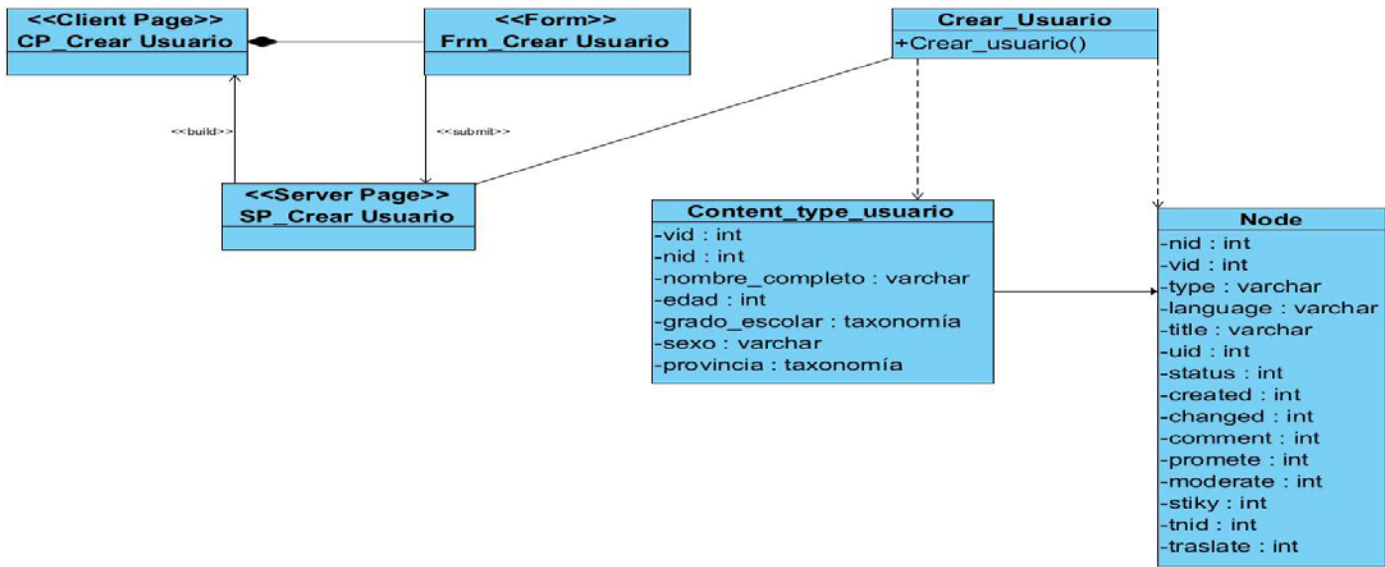


Figura 5. 14: Diagrama de clases del Diseño Crear Usuario

Diagrama de Clases de Diseño Editar Usuario

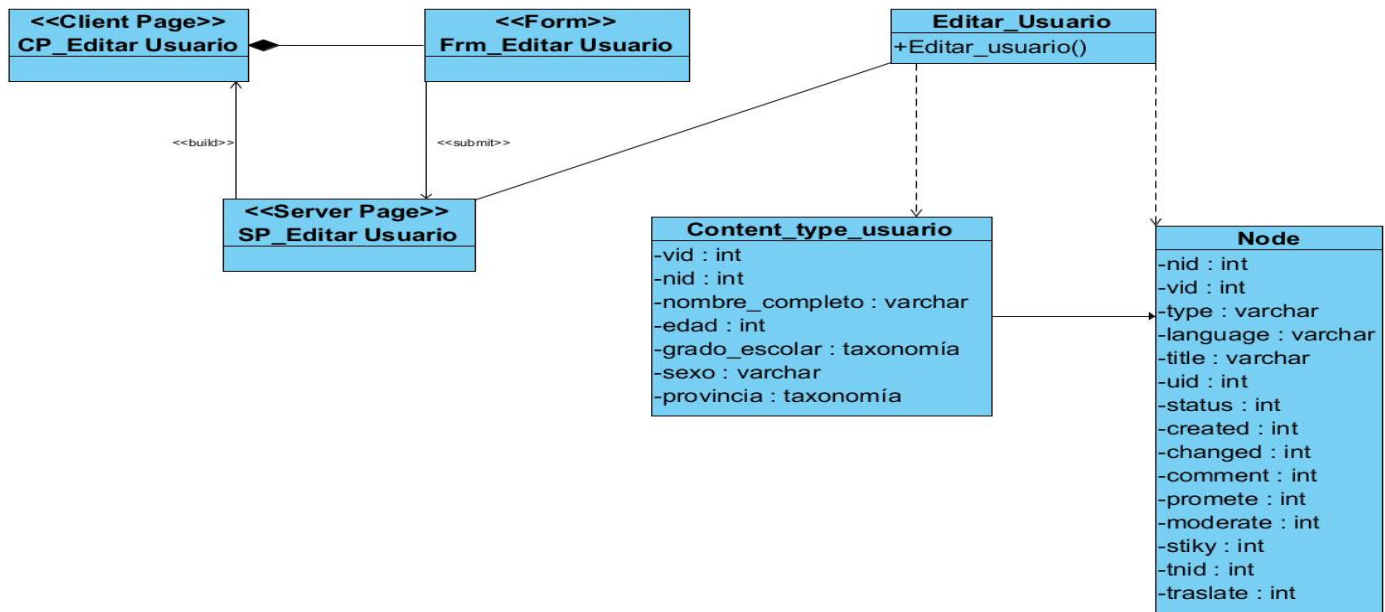


Figura 5. 15: Diagrama de clases del Diseño Editar Usuario

Diagrama de Clases de Diseño Eliminar Usuario

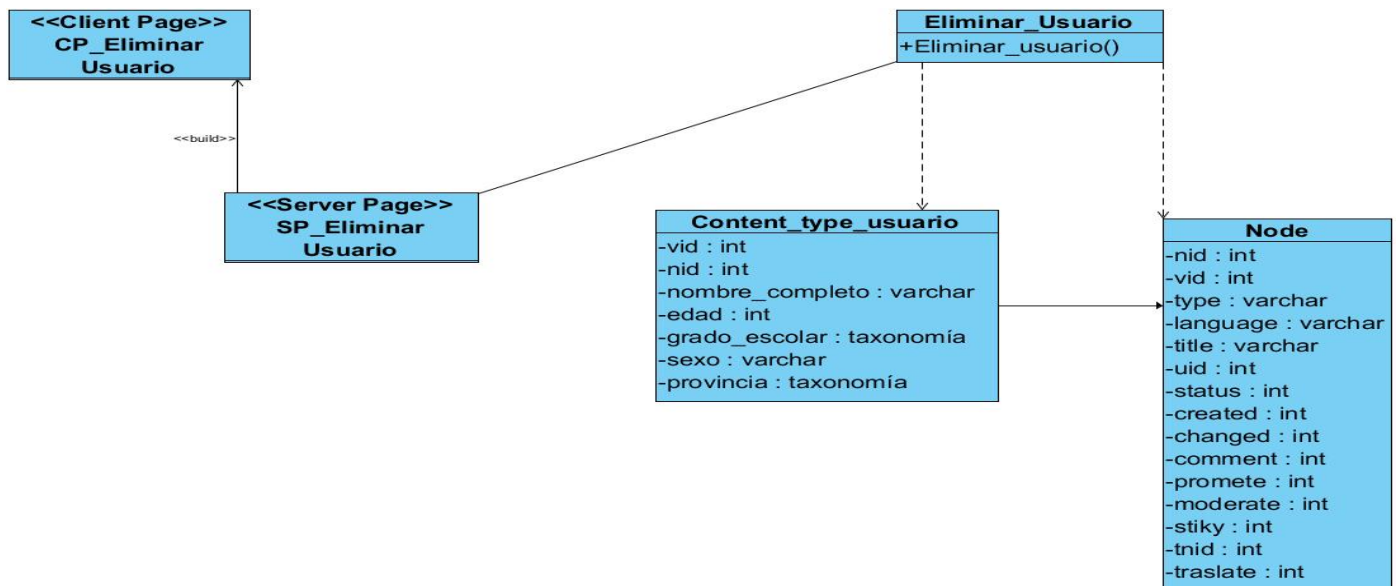


Figura 5. 16: Diagrama de clases del Diseño Eliminar Usuario

Diagrama de Clases de Diseño Listar Usuario

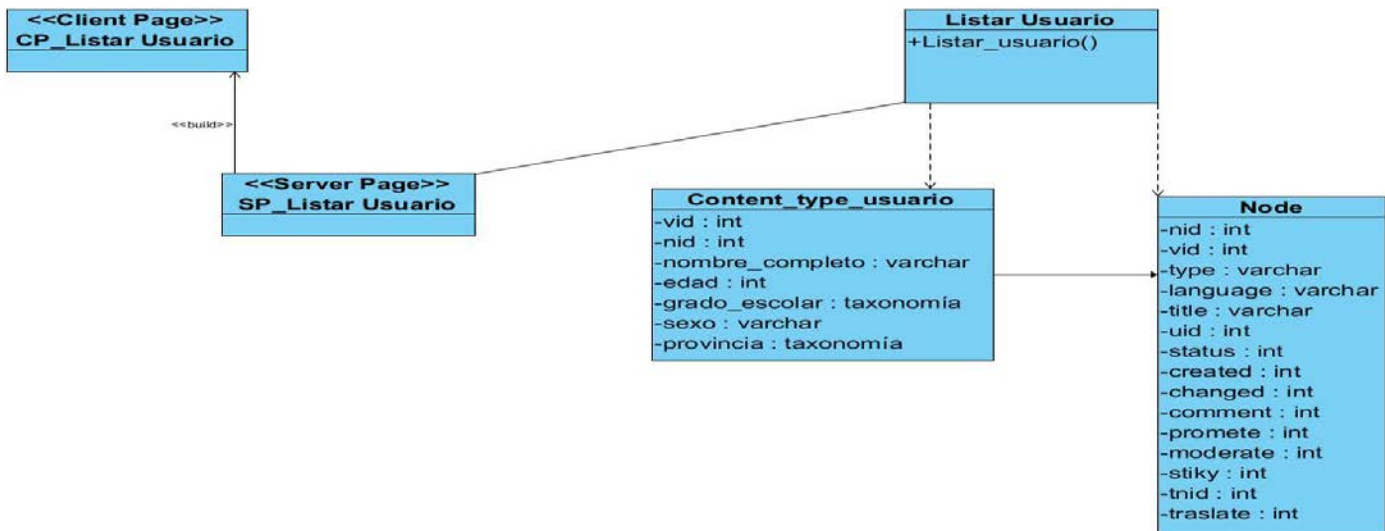


Figura 5. 17: Diagrama de clases del Diseño Listar Usuario

Diagrama de Clases de Diseño Mostrar Perfil Usuario

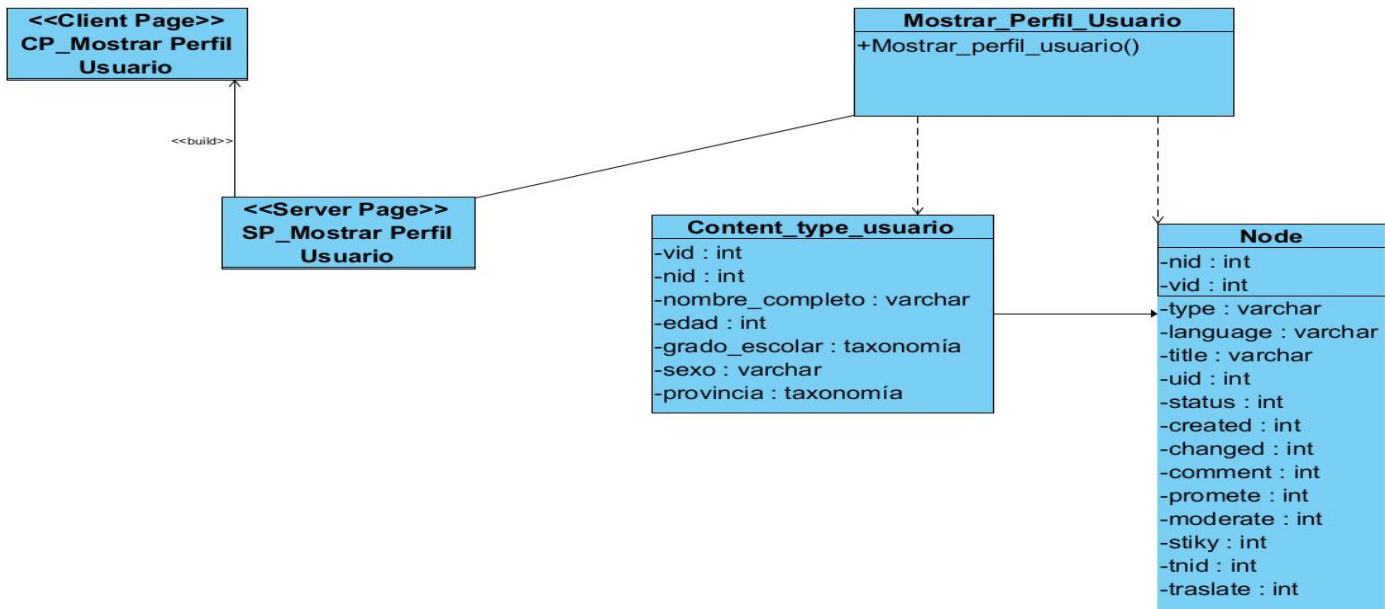


Figura 5. 18: Diagrama de clases del Diseño Mostrar Perfil Usuario

5.2 Anexos 2

Diagrama de Secuencia Editar Artículo

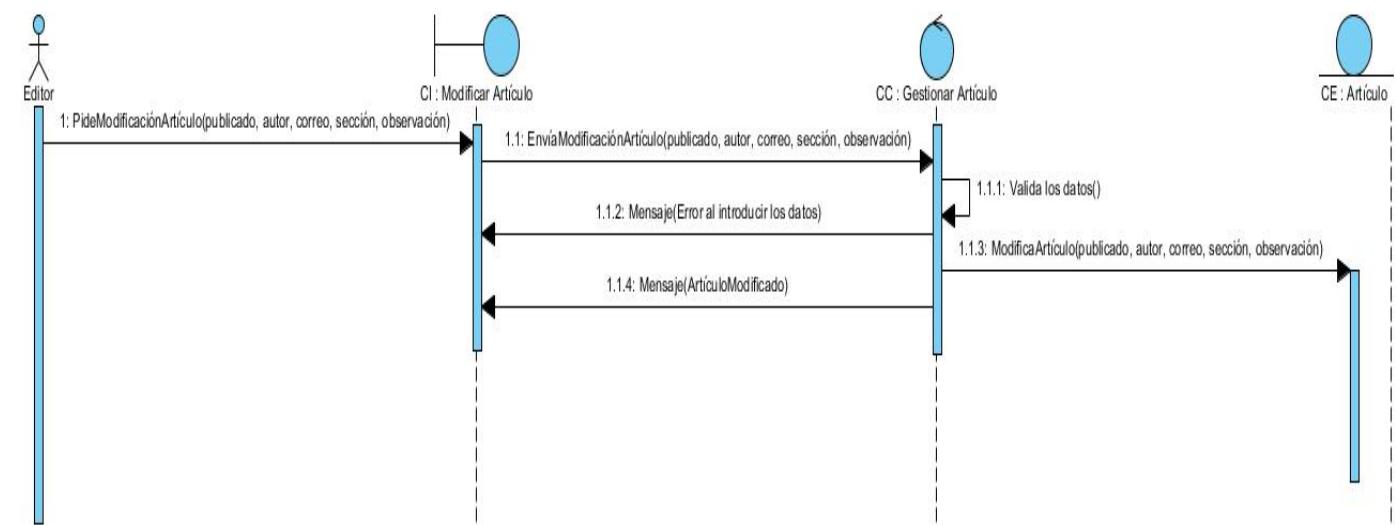


Figura 5. 19: Diagrama de Secuencia Editar Artículo

Diagrama de Secuencia Eliminar Artículo

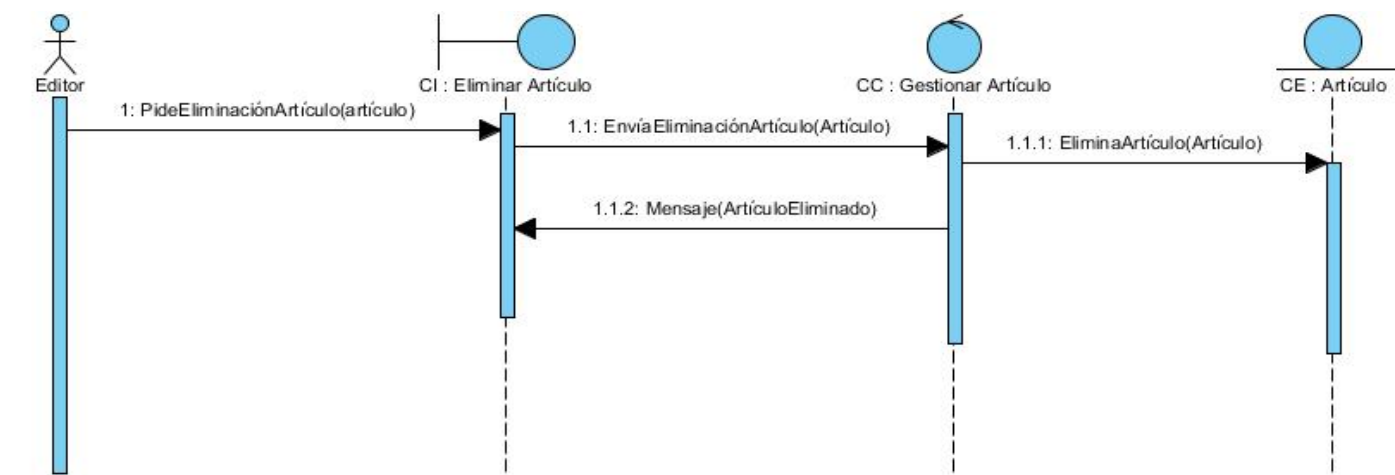


Figura 5. 20: Diagrama de Secuencia Eliminar Artículo

Diagrama de Secuencia Listar Artículo

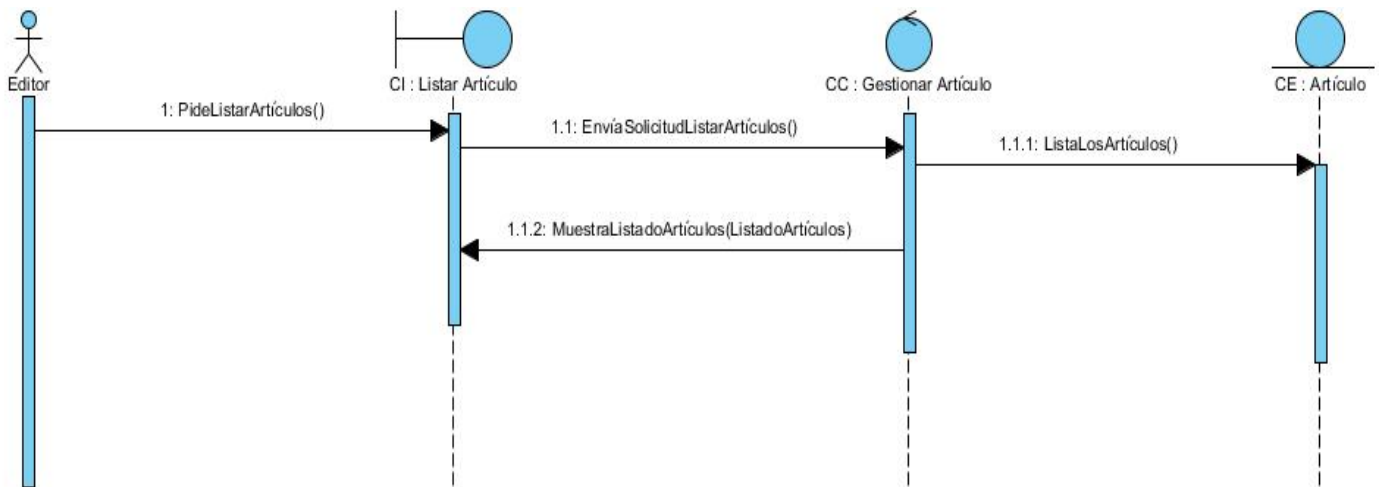


Figura 5. 21: Diagrama de Secuencia Listar Artículo

Diagrama de Secuencia Editar Concurso

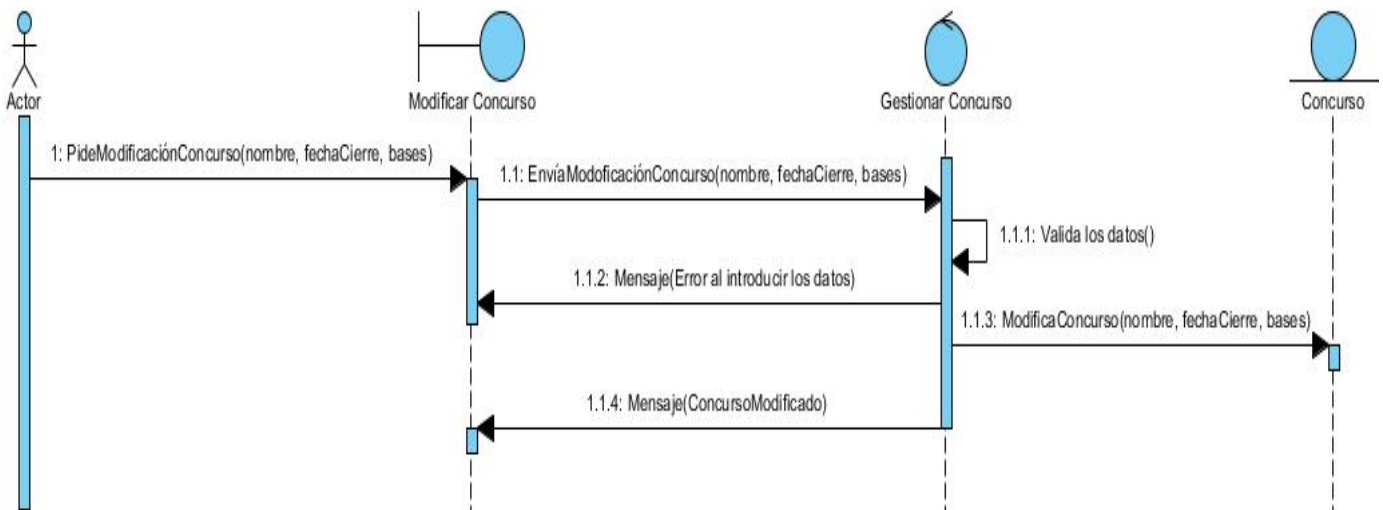


Figura 5. 22: Diagrama de Secuencia Editar Concurso

Diagrama de Secuencia Eliminar Concurso

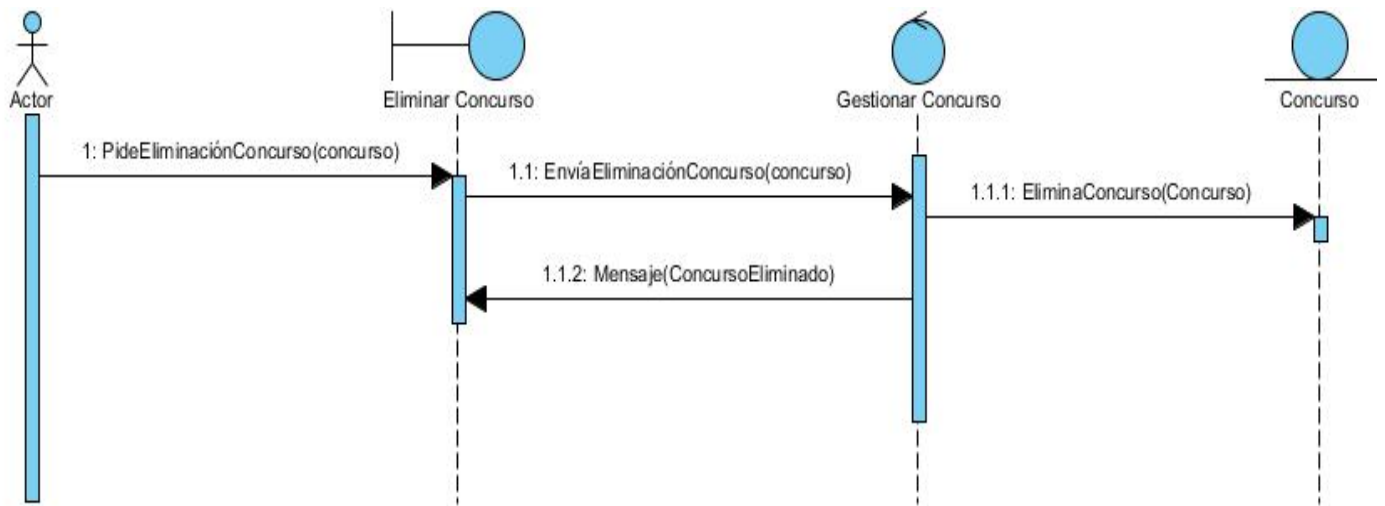


Figura 5. 23: Diagrama de Secuencia Eliminar Concurso

Diagrama de Secuencia Listar Concurso

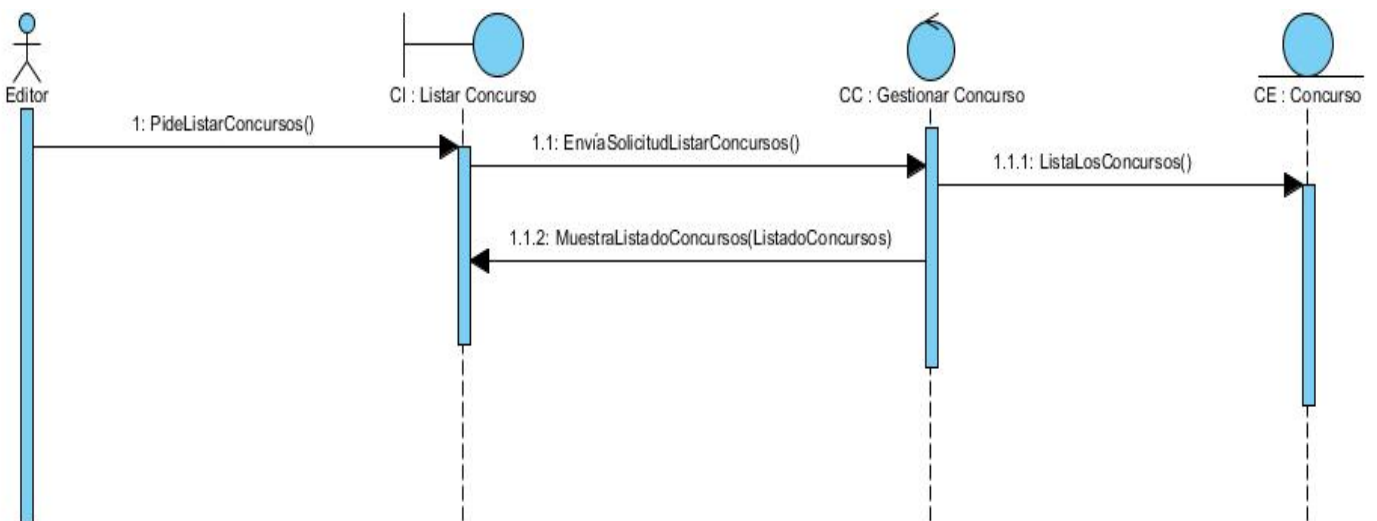


Figura 5. 24: Diagrama de Secuencia Listar Concurso

Anexos

Diagrama de Secuencia Crear Firma

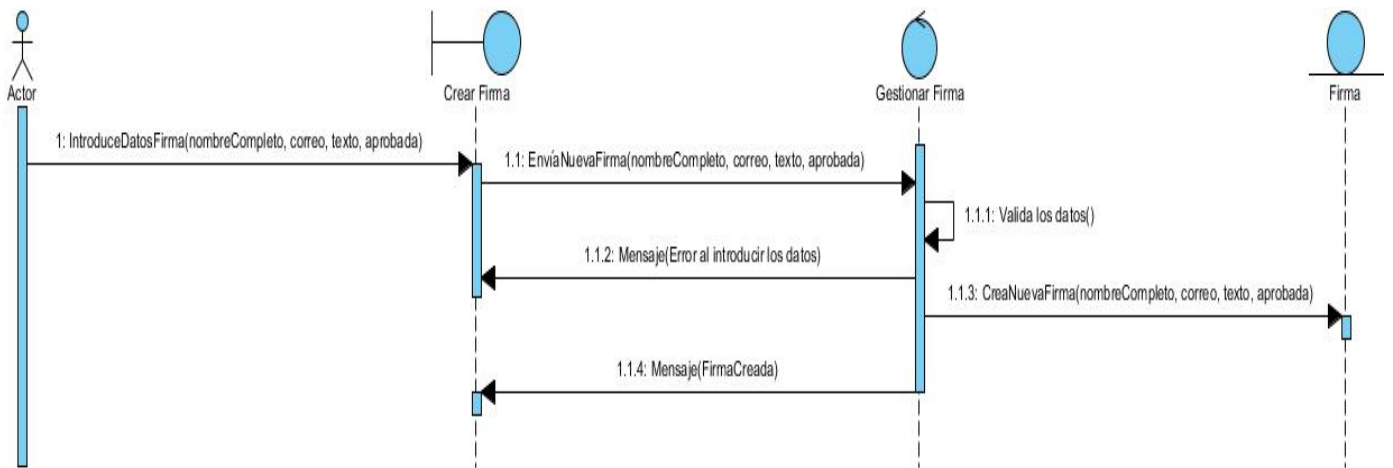


Figura 5. 25: Diagrama de Secuencia Crear Firma

Diagrama de Secuencia Editar Firma

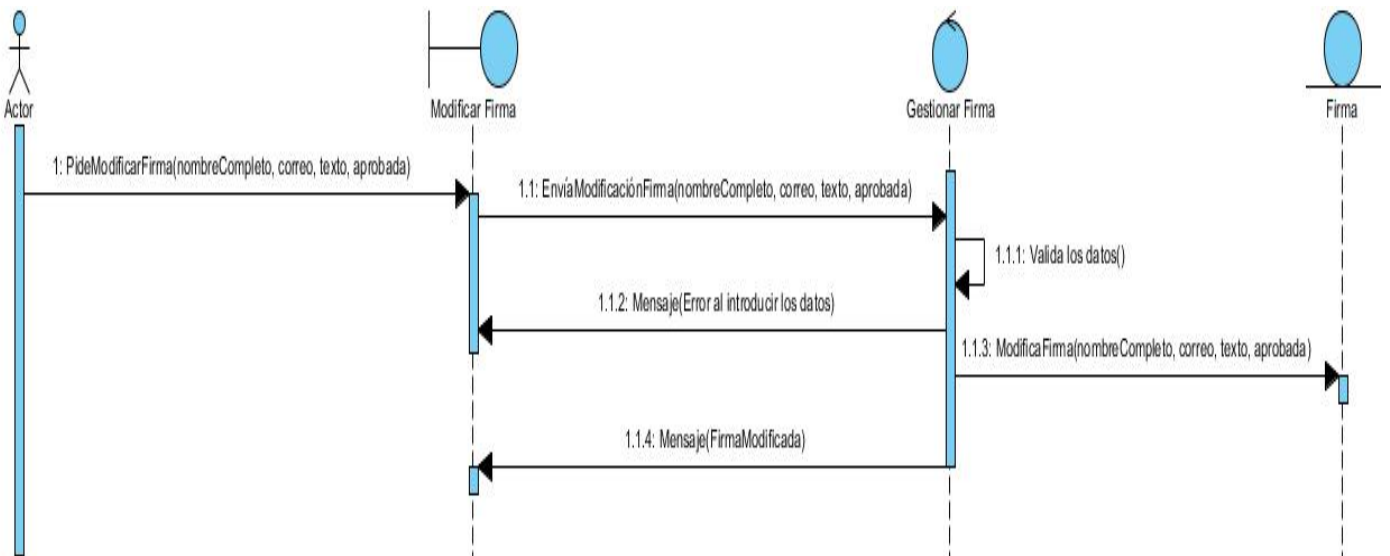


Figura 5. 26: Diagrama de Secuencia Editar Firma

Diagrama de Secuencia Eliminar Firma

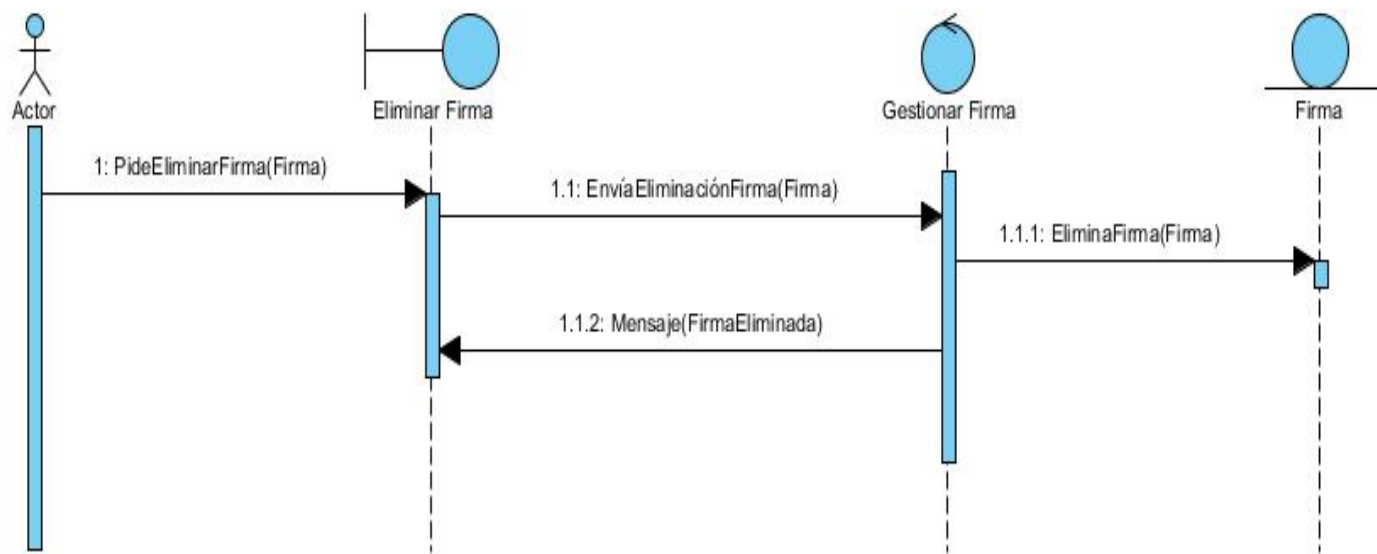


Figura 5. 27: Diagrama de Secuencia Eliminar Firma

Diagrama de Secuencia Listar Firma

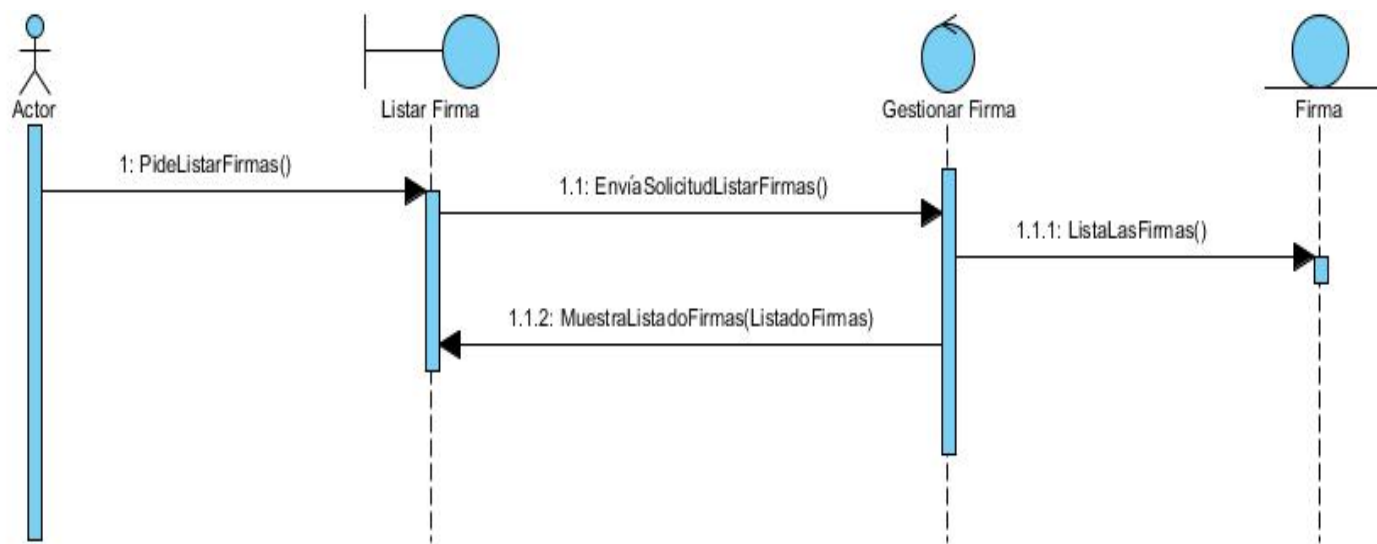


Figura 5. 28: Diagrama de Secuencia Listar Firma