

**Universidad de las Ciencias Informáticas
FACULTAD 6**



Título: Visor Web de noticias de la Plataforma de Televisión Informativa
PRIMICIA.

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

Autor(a): Lisandra Hechavarría Borrero.

Tutor(es): Ing. Félix Iván Romero Rodríguez.

Junio, 2012

FRASE

*“Nunca desistas de un sueño. Sólo trata de ver las señales que te
lleven a él.”*

Paulo Coelho.

DECLARACIÓN DE AUTORÍA

DECLARACIÓN DE AUTORÍA

Declaro ser autor de la presente tesis y reconozco a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste firmo la presente a los 28 días del mes de junio del año 2012.

Lisandra Hechavarría Borrero

Ing. Félix Iván Romero Rodríguez.

Firma del Autor

Firma del Tutor

DATOS DE CONTACTO

DATOS DE CONTACTO

Tutor: Ing. Félix Iván Romero Rodríguez (firomero@uci.cu)

Profesor instructor graduado de Ingeniero en Ciencias Informáticas en la Universidad de las Ciencias Informáticas (UCI).

AGRADECIMIENTOS

AGRADECIMIENTOS

A mi mamá y a mi papá por todo su amor, comprensión y esfuerzo realizado en estos 5 años, gracias a ellos he podido llegar donde estoy. Gracias por sus consejos oportunos y por siempre tener confianza en mí y darme ánimos para seguir adelante.

A mis hermanas, por ser las mejores del mundo y ayudarme en todo lo que necesitaba.

A mi tío, por darme siempre apoyo y aconsejarme en todo momento.

A mis amigas y amigos de todos estos 5 años en la UCI que han sido inolvidables.

A mi tutor Félix, por toda la ayuda prestada, paciencia y la habilidad que tuvo para calmarme y darme fuerzas y ánimos para enfrentarme a cada uno de mis problemas.

A mi amiga Yulia, por darme apoyo y no olvidarse de mí aunque esté lejos.

A las chicas del 101 102, por ser como una verdadera familia.

A los integrantes del proyecto Primicia, en especial Reinel, Dayana, Dorgis por ayudarme en todo lo que necesité.

A los miembros del tribunal, por todos sus consejos y recomendaciones.

De forma general, a todos lo que de una u otra forma hicieron que este sueño se hiciera realidad.

DEDICATORIA

DEDICATORIA

Le dedico este trabajo a toda mi familia, en especial a mi mamá, por estar siempre presente en todo los momentos difíciles de mi vida.

RESUMEN

RESUMEN

En la actualidad la televisión se ha convertido en un medio de comunicación muy usado en diferentes esferas sociales, demostrando un aumento de eficiencia donde se aplique. Una de sus funciones es mantener informado a toda la población. Por su parte la televisión, lejos de ir en retroceso ha logrado aliarse a la informática para mejorar cada uno de sus procesos, posibilitando la emisión de materiales de mejor calidad. La Plataforma de Televisión Informativa PRIMICIA del Centro de Geoinformática y Señales Digitales (GEySED) en la Universidad de la Ciencias Informáticas (UCI), es un ejemplo de las ventajas que ofrece la integración de la televisión y la informática, dicho sistema permite gestionar y transmitir noticias en diferentes formatos a través de una red de televisión y haciendo uso de medios informáticos.

El presente trabajo tiene como objetivo implementar un módulo al subsistema de administración de la Plataforma de Televisión Informativa Primicia capaz de visualizar en la web todas las noticias que estén publicadas. Para lograr este objetivo se realizó un estudio de las versiones anteriores de Primicia, se definieron las herramientas, tecnologías y la metodología de desarrollo más adecuada y por último se realizaron las pruebas al módulo implementado demostrando que el mismo cumple con las funcionalidades definidas.

PALABRAS CLAVE

Noticias, Plataforma de Televisión Informativa, Subsistema

ABSTRACT

ABSTRACT

Today television has become a widely used means of communication in different social spheres, showing an increase in efficiency where applicable. One of its functions is to keep the population informed. Meanwhile television, far from going in reverse has an alliance with information technology to improve each of its processes, allowing the issue of better materials. Informative television platform Primicia Geoinformatics Center and Digital Signal (GEySED) at the University of Informatics Sciences (UCI) is an example of the advantages of the integration of television and computers, the system lets you manage and deliver news in different formats via a TV network and using information technology. This work aims to implement a module management subsystem Informative television platform Primicia capable of displaying on the web all the news that is published. To achieve this objective, a study of previous versions of Premiere, defined the tools, technologies and the most appropriate development methodology and finally made the deployed module testing showing that it meets the defined functionality.

KEYWORDS

News, Informative television platform, Subsystem

ÍNDICE DE TABLAS

ÍNDICE DE TABLAS

Tabla 1 Actores Definidos del Sistema.	22
Tabla 2 Descripción del Caso de Uso Autenticar usuario.	24
Tabla 3 Descripción del Caso de Uso Listar noticias.....	26
Tabla 4 Descripción del Caso de Uso Visualizar noticias.....	28
Tabla 5 Descripción del Caso de Uso Filtrar noticias.	30
Tabla 7 Caso de Prueba Filtrar noticias	48
Tabla 8 Caso de Prueba Listar noticias	49
Tabla 9 Caso de Prueba Visualizar noticias	51

ÍNDICE DE FIGURAS

ÍNDICE DE FIGURAS

Figura 1: Esfuerzo en actividades según fase del proyecto.	11
Figura 2: Diagrama de clase del dominio.	20
Figura 3: Diagrama de casos de uso del sistema.	23
Figura 4: Diagrama de clases del análisis del CUS: Filtrar noticia.	31
Figura 5: Diagrama de clases del análisis del CUS: Listar noticia.	31
Figura 6: Diagrama de clases del análisis del CUS: Visualizar noticia.	32
Figura 7: Diagrama de clases del análisis del CUS: Autenticar usuario.	32
Figura 8: Esquema simplificado del funcionamiento interno de Symfony2.	34
Figura 9: Diagrama de clase de diseño. CU Autenticar usuario.	37
Figura 10: Diagrama de clase de diseño. CU Listar noticia.	38
Figura 11: Diagrama de clase de diseño. CU Visualizar noticia.	39
Figura 12: Diagrama de clase de diseño. CU Filtrar noticia.	40
Figura 13 Diagrama componente. CU Listar noticia.	43
Figura 14 Diagrama componente. CU Filtrar noticia.	43
Figura 15 Diagrama componente. CU Visualizar noticia.	44
Figura 16 Diagrama componente. CU Autenticar usuario.	44
Figura 17 Modelo de despliegue.	45
Figura 18 Componentes de JQuery.	60

ÍNDICE DE CONTENIDO

ÍNDICE DE CONTENIDO

INTRODUCCIÓN.....	1
CAPÍTULO 1: FUNDAMENTOS TEÓRICOS PARA EL DESARROLLO DEL VISOR WEB DE NOTICIAS DE LA PLATAFORMA DE TELEVISIÓN INFORMATIVA PRIMICIA.....	4
1.1. Introducción.....	4
1.2. Términos asociados al problema.....	4
1.2.1 Módulo	4
1.2.2 Aplicación web	4
1.2.3 Visor web	5
1.3. Análisis del objeto de estudio	6
1.3.1 Soluciones existentes a nivel nacional.....	6
1.3.2 Soluciones existentes a nivel internacional	6
1.3.3 Plataforma de televisión informativa PRIMICIA.....	8
1.3.4 Proceso de visualización de la noticia actualmente	8
1.4 Metodologías, herramientas y tecnologías a utilizar para el desarrollo de la solución.	9
1.4.1 Tecnologías a usar para el desarrollo del sistema	9
1.4.2 Metodología de Desarrollo de Software.....	9
1.4.3 Framework de desarrollo.....	12
1.4.5 Herramientas CASE	13
1.4.6 Lenguaje Unificado de Modelado.....	14
1.4.7 Entorno de desarrollo integrado(IDE)	14
1.4.8 Servidor web.....	15
1.4.9 Gestor de Base de datos	16
1.4.10 Interfaz de usuario	16
1.5 Conclusiones Parciales.....	18
CAPÍTULO 2. VISOR WEB DE NOTICIAS DE LA PLATAFORMA DE TELEVISION INFORMATIVA PRIMICIA.	19
2.1 Introducción.....	19
2.2 Modelo de Dominio	19
2.2.1 Diagrama de clases del modelo de dominio.	19
2.3 Requisitos.....	20
2.3.1 Requisitos Funcionales	20
2.3.2 Requisitos no funcionales.....	21

ÍNDICE DE CONTENIDO

2.3.2.1	Software	21
2.3.2.2	Hardware.....	21
2.3.2.3	Apariencia	21
2.3.2.4	Usabilidad.	22
2.3.2.5	Portabilidad	22
2.4	Definición de los actores del sistema	22
2.5	Diagrama de casos de uso del sistema (CUS).	22
2.6	Descripción textual de los casos de uso del sistema.....	23
2.7	Modelo de análisis	30
2.7.1	Diagramas de clases del análisis	31
2.8	Modelo de diseño	32
2.8.2	Patrones de diseño.....	34
2.8.3	Diagramas de clases del diseño.....	36
2.9	Conclusiones Parciales.....	40
CAPÍTULO 3 VALIDACIÓN DEL VISOR WEB DE NOTICIAS DE LA PLATAFORMA DE TELEVISIÓN INFORMATIVA PRIMICIA.		42
3.1	Introducción.....	42
3.2	Modelo de Implementación.....	42
3.2.1	Diagramas de componentes	42
3.3	Modelo de Despliegue	45
3.3.1	Diagramas de despliegue.....	45
3.4	Pruebas	45
3.4.1	Pruebas de caja negra.....	46
3.4.1.1	Técnica de Partición Equivalente	46
3.5	Conclusiones Parciales.....	51
CONCLUSIONES GENERALES.....		53
REFERENCIAS BIBLIOGRÁFICAS		54
BIBLIOGRAFÍAS.....		57
ANEXOS		60
GLOSARIO DE TÉRMINOS.....		61

INTRODUCCIÓN

INTRODUCCIÓN

Las Tecnologías de la Información y las Comunicaciones se encuentran en constante desarrollo, por lo que resulta importante señalar que actualmente los usuarios tienen un mayor acceso a la información. Hoy día la comunicación no solo se realiza de forma oral sino que se desarrolla a través de otros medios como la prensa, la radio, el cine y la televisión. Estas comunicaciones se han introducido en la mayoría de los espacios en la vida del hombre, haciendo que sus utilidades sean requeridas con más fuerza.

La televisión se ha convertido en uno de los medios de comunicación más usado ya que permite la visualización de imágenes y sonido en tiempo real o pasado, de sucesos acontecidos o que acontecen en ese instante, además es una forma de diversión para la sociedad mundial. Dicho medio ha alcanzado tanto desarrollo que su expansión ha sido inevitable. Su aplicación ha abarcado casi todas las esferas sociales, demostrando un aumento de eficiencia donde se aplique, destacando un mayor acceso a la información, mayor integridad y optimización de su almacenamiento.

En Cuba el desarrollo de las comunicaciones se aplica en todas las esferas de la vida, en el sector de la salud, los servicios, la educación, las investigaciones y la gestión económica. La Universidad de las Ciencias Informáticas (UCI) constituye un motor impulsor de la aplicación de estas tecnologías en el país, pues es un centro destinado al estudio y desarrollo de la tecnología informática, la cual aprovecha las posibilidades que ofrecen las Tecnologías de la Información y Comunicaciones (TICs) para brindar servicios. Esta institución desarrolla soluciones de software que permiten automatizar muchas áreas de la esfera social como la televisión. Su primera incursión en cuanto al desarrollo de plataformas de televisión informativa fue con la creación de Señal 3, un canal de noticias que tenía como objetivo mantener a la comunidad universitaria informada acerca de todo lo acontecido.

Otra plataforma similar es Señal ACN, cuyo objetivo principal es hacer llegar noticias y otras informaciones de Cuba y el mundo a los colaboradores internacionalistas que cumplen misión en el extranjero. Luego se desarrolló la plataforma TV Energía, creada para el Ministerio de Energía y Petróleo de Venezuela. Finalmente en el Centro de Geoinformática y Señales Digitales (GEySED) en la Universidad de las Ciencias Informáticas se desarrolla el producto informático PRIMICIA, el cual permite la transmisión automática y constante de noticias acompañadas de (video, imágenes, audio y teletexto) en un canal de televisión. Dicho producto se encuentra estructurado en dos subsistemas, el de Administración y el de Transmisión. En el primero se redactan las noticias y se lleva a cabo el proceso de gestión de medias de la plataforma, el cual permite eliminarlas del sistema, adicionar sus datos, reproducirlas y modificarlas. El segundo por su parte es el encargado de visualizar todas las noticias que son redactadas en el canal de televisión.

INTRODUCCIÓN

Actualmente la visualización de las noticias en el subsistema de administración de la Plataforma de Televisión Informativa Primicia no tiene un mecanismo que permita mostrar las noticias que están publicadas y en transmisión, además de no contar con una imagen para la transmisión de las noticias que sea agradable a la vista de los usuarios. Otra limitante radica en que al no contar con dicha visualización, carece de un mecanismo de control que alerte ante errores en las noticias que se transmiten o que estén próximas a transmitirse. Esto ocurre dado que para poder analizar el contenido de la noticia es necesario autenticarse en el sistema y recorrer todo el flujo hasta el proceso que muestra los contenidos informativos de la noticia publicada.

A partir de la situación problemática descrita anteriormente se plantea como **problema a resolver**: La carencia de un mecanismo que visualice las noticias publicadas en la Plataforma de Televisión Informativa PRIMICIA no permite la identificación de errores ni de contenidos inadecuados así como el registro de las noticias publicadas que se encuentran en transmisión.

Teniendo en cuenta el problema planteado fue necesario centrarse en los procesos de visualización de la noticia en la Plataformas de Televisión Informativa PRIMICIA, lo cual constituye el **objeto de estudio** de la investigación. El **campo de acción** estará enmarcado en la visualización de las noticias publicadas en el subsistema de administración de la Plataformas de Televisión Informativa PRIMICIA. Se plantea como **objetivo general**: Implementar un módulo al subsistema de Administración capaz de mostrar en la web las noticias publicadas en la plataforma.

Definidos el objeto de estudio y el campo de acción se precisa como **Idea a defender**: Con el desarrollo del visor web de la plataforma de televisión informativa PRIMICIA se logrará la interfaz para mostrar las noticias publicadas en la misma.

Se determinó que las **tareas a realizar** para darle cumplimiento al objetivo trazado estarían dirigidas a:

1. Realizar el diseño metodológico.
2. Caracterizar las soluciones existentes en el ámbito nacional e internacional.
3. Caracterizar las herramientas y tecnologías de desarrollo.
4. Establecer los requisitos funcionales y no funcionales que cumplirá el sistema.
5. Modelar del sistema.
6. Desarrollar el sistema.
7. Realizar las pruebas de funcionalidad al módulo implementado.

INTRODUCCIÓN

Para la realización de la investigación se utilizaron **métodos científicos** que se fueron aplicando durante su desarrollo.

Teóricos:

- **Analítico – Sintético:** Este método permitió estudiar la manera en que se muestran las noticias que son redactadas en el módulo de redacción actual de PRIMICIA, permitiendo la extracción de los elementos más importantes que se relacionan con el objeto de estudio, además facilitó el estudio de las herramientas y tecnologías a utilizar.
- **Análisis histórico lógico:** Este método se utilizó para estudiar las soluciones de Primicia y sus antecedentes.
- **Modelación:** Este método se utilizó en la modelación de diagramas teniendo en cuenta la metodología de desarrollo de software seleccionada.

El presente trabajo de diploma estará dividido en 3 capítulos fundamentales en los cuales se abordará todo el contenido necesario para concluir la investigación:

Capítulo 1: Este capítulo tiene como objetivo principal la fundamentación teórica de la investigación. Para esto se definieron los conceptos relacionados con el dominio del problema que permitirán obtener un mayor conocimiento a la hora de proponer una solución a la investigación, además se realiza un estudio del arte sobre las soluciones existentes de Primicia en el ámbito nacional, por último se hace un análisis de la metodología de desarrollo, herramientas y tecnologías a emplear.

Capítulo 2: En este capítulo se describen los procesos actuales a través del modelo de dominio, se seleccionan los requisitos funcionales y no funcionales que debe cumplir el sistema, los casos de uso del sistema, así como el modelo de análisis y el modelo de diseño.

Capítulo 3: En este capítulo se describe el modelo de despliegue, modelo de implementación, así como las pruebas correspondientes a la aplicación desarrollada.

Completa esta investigación un conjunto de conclusiones, los anexos que aportan información valiosa al trabajo realizado y el glosario de términos. Las fuentes bibliográficas consultadas se encuentran asentadas al final de la tesis.

CAPITULO I: FUNDAMENTACIÓN TEÓRICA

CAPÍTULO 1: FUNDAMENTOS TEÓRICOS PARA EL DESARROLLO DEL VISOR WEB DE NOTICIAS DE LA PLATAFORMA DE TELEVISIÓN INFORMATIVA PRIMICIA.

1.1. Introducción.

El objetivo de este capítulo consiste en hacer una introducción al tema de la investigación, mostrando una explicación detallada de los conceptos asociados al dominio, el análisis de las soluciones similares a Primicia, así como la identificación de las diferentes tecnologías, herramientas y lenguajes a utilizar en todo el proceso de desarrollo.

1.2. Términos asociados al problema

1.2.1 Módulo

Un módulo es una pieza o un conjunto unitario de piezas que en una construcción se repiten para hacerla más sencilla, regular y económica. El módulo, por lo tanto forma parte de un sistema y mantiene algún tipo de relación o vínculo con el resto de los componentes. (1)

Según el DRAE

Pieza o conjunto unitario de piezas que se repiten en una construcción de cualquier tipo, para hacerla más fácil, regular y económica.(2)

En programación, un módulo es un fragmento de un programa que se desarrolla de forma independiente del resto del programa.(3)

El módulo constituye un segmento importante en el desarrollo de aplicaciones, ya que va a permitir de forma general una organización de todas las funcionalidades del proyecto.

1.2.2 Aplicación web

Es una aplicación informática distribuida cuya interfaz de usuario es accesible desde un cliente web, normalmente un navegador web.(4)

Aplicaciones que los usuarios pueden utilizar accediendo a un servidor web a través de internet o de una intranet mediante un navegador. Es decir, es una aplicación software que se codifica en un lenguaje soportado por los navegadores web en la que se confía la ejecución al navegador. Las

CAPITULO I: FUNDAMENTACIÓN TEÓRICA

aplicaciones web son populares debido a la facilidad para actualizar y mantenerlas sin tener que distribuir e instalar software a miles de usuarios. (5)

Dichas aplicaciones presentan algunas características fundamentales que son útiles para su servicio las cuales son:

- Comunicación mediante HTTP¹ sobre TCP²/IP³.
- Procesamiento en servidor
- Acceso a base de datos.
- Arquitectura por capas.
- Distintos tipos de usuarios.

Las aplicaciones web constituyen un aspecto fundamental en cualquier sistema ya que son las encargadas de mantener la comunicación entre los usuarios y un servidor.

1.2.3 Visor web

Tipo de aplicación utilizada para visualizar algún archivo o recurso como imágenes, noticias, multimedia y texto. (6)

El desarrollo de visores web hoy día se ve reflejado en diversas funciones, tanto para divulgar algún contenido como para visualizarlos. Es por eso que en diferentes partes del mundo el uso de los visores web es de mucha ayuda y tiene gran importancia. Principalmente están destinados a comportarse como multimedia, mostrar informaciones geográficas permitiendo consultar servicios estándares de mapas y mostrar imágenes en la web, los cuales pertenecen a una categoría de software donde la cantidad y calidad de programas gratis que existen hace difícil seleccionar solo un producto como elección principal. Por eso la conveniencia personal aparecerá inevitablemente como el factor definitivo para esta comparativa de aplicaciones gratuitas. Los mismos son muy utilizados en los sitios web, los cuales le ofrecen un valor agregado a los mismos.

¹ HTTP (Protocolo de Transferencia de Hipertextos)

² TCP (Protocolo de Control de Transmisión).

³ IP (Protocolo de Internet)

CAPITULO I: FUNDAMENTACIÓN TEÓRICA

1.3. Análisis del objeto de estudio

Este epígrafe abordará acerca de las soluciones existentes similares a Primicia que han sido desarrolladas y se analizarán sus principales características.

1.3.1 Soluciones existentes a nivel nacional

Señal ACN⁴, sistema automatizado de teletexto para la plataforma de televisión digital satelital cubana, constituye una plataforma con el objetivo de acercar la información a aquellos colaboradores cubanos, que prestan servicios fuera de cuba, además de llegar a las zonas de silencio, lugares donde la actualidad se abría paso paulatinamente. Señal ACN no tiene concebido los nuevos elementos que se desean integrar al proceso de redacción de PRIMICIA. Este sistema no permite la configuración de los elementos que componen la noticia.(7)

Señal3: Sistema de producción de noticias desarrollado en la Universidad de las Ciencias Informáticas. Diseñado para mantener informado a la comunidad universitaria. Este sistema fue desarrollado sobre tecnología privativa como base.(7)

TV Energía: Resultado devenido de la investigación sobre el estudio de sistemas informáticos sobre tecnologías libre para la transmisión. Este sistema fue destinado a automatizar los procesos que se realizan para visualizar la información en la red de televisión instalada en el Ministerio del Poder Popular para la Energía y Petróleo de Venezuela. (7)

Estos sistemas nacionales constituyen los sistemas similares a PRIMICIA desarrollados anteriormente, estos productos cuentan con la posibilidad de acceso, actualización y presentación de los recursos de multimedia en el proceso de redacción de la noticia, sin embargo ninguno de ellos tiene la posibilidad de mostrar las noticias publicadas en una interfaz web de manera que el usuario verla antes de ser transmitida.

1.3.2 Soluciones existentes a nivel internacional

En el orbe existen varias empresas como televisoras, que hacen uso de las utilidades que facilitan la tecnología de los sistemas de producción de noticias, confirmando con la experiencia la factibilidad que provee. Entre empresas que han incursionado en la misma se encuentran:

4 ACN: Agencia Cubana de Noticias

CAPITULO I: FUNDAMENTACIÓN TEÓRICA

BBC⁵: esta lanzó BBC News 24, esta logró una completa digitalización y una nueva estructura de los servicios informativos, transformando los modelos que conservaban la brecha que se sostenía entre las vías de redacción principal del canal, las de radio e Internet. Esta plataforma cuenta con un sistema capaz de automatizar transmisiones en directo, la redacción dispone de cuarenta y siete terminales en las que el periodista puede editar. Este sistema permite transferir clips de noticias producidos por el canal de 24 horas a los servidores de la redacción.(8)

SONAPS: es un entorno en red que integra todos los aspectos del proceso de producción de informativos, desde la planificación, captación y edición, hasta la emisión, distribución y archivo. Permite que periodistas, editores profesionales y productores puedan compartir contenido y metadatos en un colaborativo flujo de trabajo en red basado en ficheros. El sistema Sonaps transforma el proceso de producción de noticias integrando todos los aspectos del proceso de producción en un completo y sencillo flujo de trabajo. Una vez ya se han acordado y planificado las historias, Sonaps se encarga de todo desde la búsqueda de contenido en el Archivo hasta la gestión de los datos de planificación para los reporteros.(9)

En la sala de Redacción, Sonaps se integra por completo con los sistemas de informativos, desde la etapa de planificación pasando por la edición y emisión hasta el archivo. Los contenidos dentro de Sonaps y en sistemas de archivo profundos pueden ser buscados y visionados fácilmente.(10)

ENEWSROOM: es un sistema de producción de noticias y contenidos audiovisuales diseñado para facilitar el trabajo de periodistas y usuarios. Este sistema permite una redacción desde varios clientes, los cuales son centralizados en un servidor central, de manera que se publican las noticias de cada redactor. Este sistema es usado principalmente en empresas comerciales.(11)

Entre sus características principales se encuentra (12):

- Sistema sencillo, intuitivo, robusto y fiable.
- Posee servidores que son capaces de digitalizar simultáneamente hasta tres calidades de vídeo.
- La base del sistema se conecta con otras herramientas externas de la plataforma Estructure que aportan sus funcionalidades a la redacción.

5 BBC: British Broadcasting Company por sus siglas en Inglés. Corporación Británica de Radiodifusión

CAPITULO I: FUNDAMENTACIÓN TEÓRICA

- Utiliza metadatos personalizables donde los contenidos multimedia pueden ser etiquetados con términos descriptivos pertinentes para que los editores puedan localizarlos de forma rápida.

1.3.3 Plataforma de televisión informativa PRIMICIA

PRIMICIAv1.6, es un producto informático desarrollado en el centro productivo Geoinformática y Señales Digitales (GEYSED), específicamente en el Departamento de señales digitales. Es una solución integral la cual provee un canal de televisión para la transmisión automática y constante de informaciones en distintos formatos. Une varios formatos como son: texto, imagen, video y audio, con el objetivo de mostrar noticias de una forma más agradable a la vista de los usuarios. Es un sistema multiplataforma desarrollado con software libre. Esta solución puede ser ajustable a cualquier cliente que tenga la necesidad o interés de mantener informados de manera rápida y constante a un grupo significativo de personas.

Está estructurado en dos subsistemas que se relacionan entre sí y actúan como un todo para brindar un resultado eficiente y acorde a las necesidades de los usuarios:

- Subsistema de Administración.
- Subsistema de Transmisión.

El subsistema de administración permite la administración del canal y toda la gestión de las noticias y recursos multimedia, mientras que el subsistema de transmisión se encarga de visualizar las noticias y materiales publicados.

Funcionalidades de Primicia:

- Muestra de forma automática ciclos de noticias constantes y repetitivos.
- Permite la utilización de cintillos informativos o infocintas con informaciones relevantes y breves.
- Permite mostrar información adicional a la noticia (fecha y hora, próxima noticia y otras).
- Permite la transmisión de señales de video externas.

1.3.4 Proceso de visualización de la noticia actualmente

El subsistema de administración del producto Primicia es el encargado de toda la gestión de las noticias, el mismo está dividido en diferentes módulos. Uno de los más importantes es el de redacción de noticias, en el mismo se realiza la creación y edición de las informaciones y los recursos multimedia

CAPITULO I: FUNDAMENTACIÓN TEÓRICA

que van a conformar la noticia. Para acceder a dicho módulo lo primero que debe hacer el usuario es autenticarse, una vez autenticado tiene la posibilidad de acceder al modulo de redacción para crear la noticia en la que debe establecer un título, ubicarla en una sección temática, elegir el fondo musical y crear las pantallas que la conforman las cuales pueden ser de tipo texto, texto-imagen, imagen y video. Una vez que la noticia es redactada en dicho módulo y se realizan todas las acciones que esta tiene se genera de forma automática una pre visualización de la noticia con la sección a la que pertenece, título, fecha, fondo musical y el tipo de pantalla que puede ser de tipo texto, texto-imagen, imagen y video. Además tiene las opciones de editar, eliminar y nueva pantalla. Todas estas acciones se realizan dentro del módulo de redacción, lo que trae como consecuencia que para visualizar las noticias que están publicadas el usuario tiene que tener los permisos necesarios para acceder al módulo y poder ver la noticia. Por lo que es necesario que exista una manera en que el usuario sin entrar al módulo redacción pueda ver las noticias que están publicadas. Posibilitando que en caso de que las noticias contengan determinados errores en su contenido o se desee cambiar se tenga la posibilidad de realizarlo antes de que esta sea transmitida por el canal de transmisión.

1.4 Metodologías, herramientas y tecnologías a utilizar para el desarrollo de la solución.

1.4.1 Tecnologías a usar para el desarrollo del sistema

Para desarrollar una aplicación es necesario contar con tecnologías que permitan que el ciclo de desarrollo del software sea un proceso exitoso. Las mismas deben facilitar la implementación, permitir compatibilidad y ofrecer posibilidades de generar un software escalable y potente. En este caso para el desarrollo de la aplicación se utilizará algunas tecnologías que fueron empleadas en la versión 1.5 de Primicia.

1.4.2 Metodología de Desarrollo de Software

Todo desarrollo de un software generalmente se guía por un procedimiento que le permita de manera correcta cumplir con el propósito del producto a realizar. Por lo que la metodología de desarrollo es el pilar fundamental en el desarrollo de cualquier software ya que se guía por conjunto de procedimientos, técnicas, herramientas y un soporte documental que ayuda a los desarrolladores a realizar nuevo software. En el proceso de desarrollo de software la metodología define Quién debe hacer Qué, Cuándo y Cómo debe hacerlo para obtener los distintos productos parciales y finales.

CAPITULO I: FUNDAMENTACIÓN TEÓRICA

Existen diferentes tipos de metodologías de desarrollo de software, una de ellas es RUP⁶ siendo una de las metodologías tradicionales más usada en la universidad.

Se escogió esta metodología ya que es la más usada en la escuela, además la aplicación que se va a desarrollar está estrictamente vinculada a un proyecto. Por lo que la documentación que se genera debe ser compatible con la existente.

El **Proceso Unificado Racional (RUP)** es un proceso de desarrollo de software y junto con el Lenguaje Unificado de Modelado UML, constituye la metodología estándar más utilizada para el análisis, implementación y documentación de sistemas orientados a objetos. (13)

Provee a cada miembro del equipo, un fácil acceso a una base de conocimiento con guías, plantillas y herramientas para todas las actividades críticas del desarrollo de software. Esta metodología permite que todos los integrantes de un equipo de trabajo, conozcan y compartan el proceso de desarrollo, una base de conocimientos y los distintos modelos de cómo desarrollar el software. Su principal objetivo es asegurar la producción de software de alta calidad, que cumpla las necesidades de sus usuarios finales, que sea realizado en las fechas acordadas y con el presupuesto disponible. (14)

RUP presenta 3 características que constituyen la esencia de todo el proceso de desarrollo:

1. **Dirigido por los casos de uso:** Que orientan el proyecto a la importancia para el usuario y lo que este quiere.(15)
2. **Centrado en la arquitectura:** Que relaciona la toma de decisiones que indican cómo tiene que ser construido el sistema y en qué orden.(15)
3. **Ciclo de vida iterativo e incremental:** Donde divide el proyecto en mini proyectos donde los casos de uso y la arquitectura cumplen sus objetivos de manera más depurada. (15)

Otras características o ventajas de la aplicación de esta metodología son las siguientes:

- Reconoce que las necesidades del usuario y sus requerimientos no se pueden definir completamente al principio.

⁶RUP: Proceso Unificado de Desarrollo

CAPITULO I: FUNDAMENTACIÓN TEÓRICA

- Permite evaluar tempranamente los riesgos en lugar de descubrir problemas en la integración final del sistema.
- Facilita la reutilización del código teniendo en cuenta que se realizan revisiones en las primeras iteraciones lo cual además permite que se aprecien oportunidades de mejoras en el diseño.

El proceso de desarrollo está dividido en fases a lo largo del tiempo cada una de las cuales tiene objetivos específicos y un conjunto de artefactos definidos que deben alcanzarse. La duración de cada fase depende del equipo y del producto a generar, las mismas pasan por los distintos flujos de trabajo.

A su vez, cada fase puede tener una o más iteraciones y cada iteración sigue el modelo en cascada pasando por las distintas disciplinas. Cada iteración termina con una liberación del producto.

Las fases son las siguientes(15)

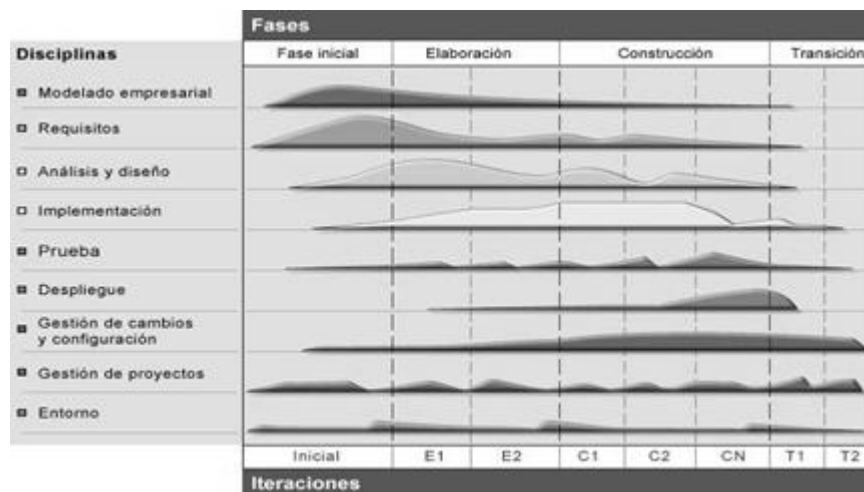


Figura 1: Esfuerzo en actividades según fase del proyecto.

Inicio: se hace un plan de fases, se identifican los principales casos de uso y se identifican los riesgos. Se define el alcance del proyecto, y se pone mayor énfasis en actividades modelado del negocio y de requisitos.

Elaboración: se hace un plan de proyecto, se completan los casos de uso y se eliminan los riesgos.

Construcción: se concentra en la elaboración de un producto totalmente operativo y eficiente y el manual de usuario.

CAPITULO I: FUNDAMENTACIÓN TEÓRICA

Transición: se instala el producto en el cliente y se entrena a los usuarios. Como consecuencia de esto suelen surgir nuevos requisitos a ser analizados.

1.4.3 Framework de desarrollo

Además de los lenguajes antes mencionados, resulta importante mencionar un framework que resalta por sus ventajas en el desarrollo de aplicaciones web: Este es, Symfony en su versión 2.0:

Symfony 2.0:

Symfony2 ha sido ideado para aprovechar las características del lenguaje PHP 5.3. Su arquitectura interna está completamente desacoplada, lo que permite reemplazar o eliminar fácilmente aquellas partes que no se ajustan al proyecto. Facilita el desarrollo de las aplicaciones web, se encarga de todos los aspectos comunes de las aplicaciones web, dejando que el programador se dedique a aportar valor desarrollando las características únicas de cada proyecto. (16) Emplea el tradicional patrón de diseño MVC (modelo-vista-controlador) para separar las distintas partes que forman una aplicación web. EL modelo representa la información con la que trabaja la aplicación y se encarga de acceder a los datos, divide los proyectos web en aplicaciones y módulos (17)

Por su importancia en el siguiente trabajo se muestran algunas de sus características más importantes:

Características(18)

- **Usabilidad avanzada:** Es un framework fácil de utilizar gracias a que cuenta con una interfaz de aplicación de desarrollo muy intuitiva.
- **Extensible:** Se construye a base de bundles⁷.
- **Construido para desarrolladores:** Proporciona las herramientas que en gran medida mejoran la productividad de los desarrolladores, como la famosa barra de depuración web, soporte nativo de entornos, páginas detalladas de errores y mucho más.

Symfony2 cuenta con todas las características que el desarrollador de aplicaciones web necesita como la de permitir integrarse con otras tecnologías, que permiten que las aplicaciones se

⁷Bundle: Es una carpeta que contiene los archivos necesarios para un grupo de funcionalidades específicas, como por ejemplo un blog, un carrito de compras.

CAPITULO I: FUNDAMENTACIÓN TEÓRICA

desarrollen con mayor facilidad. También proporciona seguridad integrada y promueve el desarrollo web utilizando buenas prácticas.

1.4.4 Lenguajes de programación

JavaScript: lenguaje de programación interpretado, basado en prototipos que permite a los desarrolladores crear acciones en sus páginas web. No requiere de compilación ya que el lenguaje funciona del lado del cliente y los navegadores son los encargados de interpretar el código. Es soportado por la mayoría de los navegadores como Internet Explorer y Mozilla Firefox. Tiene la ventaja de ser incorporado en cualquier página web y puede ser ejecutado sin la necesidad de instalar otro programa para ser visualizado. (19)

HTML⁸ (Lenguaje de Marcado de Hipertexto): es el lenguaje que se utiliza para crear las páginas web, se basa en el uso de etiquetas. (20) Este lenguaje indica a los navegadores cómo deben mostrar el contenido de una página web luego de interpretarlo. Los documentos del mismo son ficheros de texto plano que pueden ser editados con cualquier editor de texto. Se basa en la sintaxis SGML (*Standard Generalized Markup Language*). Esto quiere decir que los diferentes elementos (párrafos, encabezamientos, tablas, listas, de un documento para la WWW se señalan intercalando etiquetas que indican al navegador cómo debe mostrarlo.(21)

PHP⁹: Es un lenguaje interpretado de alto nivel embebido en páginas HTML, ejecutado en el servidor y de código abierto. La meta del lenguaje es permitir rápidamente a los desarrolladores la generación dinámica de páginas”. Una de sus características más poderosa es su soporte para gran cantidad de bases de datos, ofrece la integración con las varias bibliotecas externas, que permiten que el desarrollador haga casi cualquier cosa desde generar documentos en pdf hasta analizar código XML¹⁰. (22)

1.4.5 Herramientas CASE

Visual Paradigm: es una herramienta multiplataforma, soporta el ciclo de vida completo del desarrollo de software: análisis y diseño orientados a objetos, construcción, pruebas y despliegue. El software de

⁸HTML: Lenguaje de Marcado de Hipertexto

⁹PHP: Preprocesador de hipertexto

¹⁰XML: Lenguaje Extensible de Marcas

CAPITULO I: FUNDAMENTACIÓN TEÓRICA

modelado UML ayuda a una más rápida construcción de aplicaciones de calidad y a un menor costo. Posibilita la elaboración de diferentes diagramas como: diagrama de casos de uso, diagramas de clases, diagrama de componentes, entre otros. Se integra con varios Entornos de Desarrollo Integrado. Funciona también como herramienta colaborativa, soportando un equipo de trabajo sobre un mismo proyecto, incluyendo la capacidad del control de versiones, emplea una rápida respuesta con poca memoria utilizando moderadamente los tiempos del procesador, lo que le permite manejar grandes y complicadas estructuras de un proyecto en una forma muy eficiente y, que solo requiera de una configuración de escritorio. (23)(24)

1.4.6 Lenguaje Unificado de Modelado

El **Lenguaje de Modelado Unificado (UML)** es un lenguaje para la especificación, visualización, construcción y documentación de los artefactos de un proceso de sistema intensivo. Es de gran ayuda para los analistas y arquitectos de sistemas que trabajan en el diseño y análisis de objetos de un lenguaje, así también es útil para hacer modelos de negocios.(25)

Está pensado para usarse con todos los métodos de desarrollo, etapas del ciclo de vida, dominios de aplicación y medios. El lenguaje de modelado pretende unificar la experiencia pasada sobre técnicas de modelado e incorporar las mejores prácticas actuales en un acercamiento estándar. Está dado de múltiples herramientas para lograr la especificación determinante del modelo, pero en el presente trabajo se trabaja en forma simplificada sobre: Modelado de Clases, Casos de Uso y Diagramas. Incrementa la capacidad de lo que se puede hacer con otros métodos de análisis y diseño orientados a objetos. (26) Dicho lenguaje es el usado por la herramienta CASE visual parading para la elaboración de los diferentes diagramas que se desean desarrollar.

1.4.7 Entorno de desarrollo integrado(IDE)¹¹

IDE de desarrollo: 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 (22). Además de ser aplicaciones claves para una implementación rápida y eficaz del software, los mismos se deben seleccionar en dependencia de los lenguajes en que se piensa desarrollar, puesto que no todos soportan los lenguajes requeridos.(28)

¹¹IDE: Entorno de Desarrollo Integrado

CAPITULO I: FUNDAMENTACIÓN TEÓRICA

NetBeans: Es un entorno de desarrollo integrado (IDE) multiplataforma, gratuito, de código abierto para desarrolladores, libre y sin restricciones de uso. (29) Tiene una estructura para los proyectos que se pueden crear junto a este IDE. Propone un esqueleto para organizar código fuente, el editor conjuntamente integra los lenguajes como HTML, Java Script y CSS. Tiene la posibilidad de contar con la integración de sistemas de control de versiones, tales como RapidSVN usado para mantener el proyecto guardado en el repositorio. Dando la posibilidad de actualizar y enviar los cambios realizados en la aplicación que se desarrolla.

1.4.8 Servidor web

Un **servidor web** es un programa que se ejecuta continuamente en un computador, manteniéndose a la espera de peticiones de ejecución que le hará un cliente o un usuario de Internet. El servidor web se encarga de contestar a estas peticiones de forma adecuada, entregando como resultado una página web o información de todo tipo de acuerdo a los comandos solicitados.

Apache: El servidor HTTP Apache, es un software libre, de código abierto y multiplataforma. Por su excelencia, robustez y estabilidad hacen que cada vez millones de servidores reiteren su confianza en este programa. Usa la licencias BSD¹², la cual permite realizar cambios al código fuente. Actualmente está en la versión 2.0 la cual es la que se usará en la aplicación a desarrollar. Dicha versión posee las siguientes características:

Características(30)(31)

- Modular: Puede ser adaptado a diferentes entornos y necesidades con los diferentes módulos de apoyo que proporciona.
- Extensible: gracias a ser modular se han desarrollado diversas extensiones entre las que destaca PHP, un lenguaje de programación del lado del servidor.
- Trabaja con gran cantidad PHP.
- Permite personalizar la respuesta ante los posibles errores que se puedan dar en el servidor.
- Nuevo sistema de configuración y compilación: El sistema de configuración y compilación ha sido escrito de nuevo desde haciendo que se parezca ahora más al de otros proyectos código abierto.

¹²Licencia Berkeley Software Distribution (BSD) es la licencia de software otorgada principalmente para los sistemas.

1.4.9 Gestor de Base de datos

Un **Sistema Gestor o Manejador de Bases de Datos (SGBD)** es un conjunto de programas que permite a los usuarios crear y mantener una base de datos, por lo tanto, el SGBD es un software de propósito general que facilita el proceso de definir, construir y manipular la base de datos para diversas aplicaciones. Estos pueden ser de propósito general o específico.

PostgreSQL:

Es un sistema de gestión de bases de datos objeto-relacional. Se ejecuta en la mayoría de los Sistemas Operativos más utilizados en el mundo incluyendo, Linux, varias versiones de UNIX y en Windows.(32)

Características de PostgreSQL (33)

- Soporta distintos tipos de datos: datos de tipo fecha, monetarios, elementos gráficos, datos sobre redes, cadenas de bits. También permite la creación de tipos propios.
- Incluye herencia entre tablas (aunque no entre objetos, ya que no existen), por lo que a este gestor de bases de datos se le incluye entre los gestores objeto-relacionales.
- Orientado a objetos.

1.4.10 Interfaz de usuario

Interfaz de Usuario: Para que el usuario interactúe a gusto con un sistema necesita, como aspecto primordial, de una interfaz de usuario atractiva que lo motive y le sea entendible e intuitiva en el trabajo que realiza. Para el desarrollo de la misma en tecnología web, se emplean librerías que facilitan este trabajo, aunque existen también métodos tradicionales de diseño que emplean imágenes estáticas o animadas y textos decorados.

Librería JQuery

JQuery: librería de Java Script pensada para interactuar con los elementos de una web por medio del Modelo de objeto del documento (DOM) de un modo simplificado. Ofrece una infraestructura con la que se tiene mucha facilidad para la creación de aplicaciones complejas del lado del cliente. Por ejemplo, con JQuery se obtiene ayuda en la creación de interfaces de usuario, efectos dinámicos,

CAPITULO I: FUNDAMENTACIÓN TEÓRICA

aplicaciones que hacen uso de Ajax¹³. Cuando se programa Java script con JQuery se tiene a la disposición una interfaz para la programación. Simplemente se debe conocer las librerías y programar utilizando las clases, sus propiedades y métodos para la concepción de los objetivos.(34)

Además, todas estas ventajas, con JQuery se logra de manera gratuita, ya que la librería tiene licencia para uso en cualquier tipo de plataforma, personal o comercial. Para ello simplemente se incluye un script Java script que contiene el código de JQuery.

La librería JQuery en resumen aporta las siguientes ventajas(34):

- Ahorra muchas líneas de código.
- Hace transparente el soporte de la aplicación para los navegadores principales.
- Provee de un mecanismo para la captura de eventos.
- Provee un conjunto de funciones para animar el contenido de la página en forma muy sencilla.
- Integra funcionalidades para trabajar con Ajax.
- Sencillez en su sintaxis.

Es importante comentar que JQuery no es la única librería que existe en el mercado, pero es un producto con una aceptación muy buena por parte de los programadores. Además, es un producto serio, estable, bien documentado y con un gran equipo de desarrolladores a cargo de la mejora y actualización de la librería. Otra cosa muy interesante es la dilatada comunidad de creadores de plugins o componentes, lo que hace fácil encontrar soluciones ya creadas en JQuery para implementar asuntos como interfaces de usuario, galerías y votaciones

Por lo planteado anteriormente y el análisis riguroso entre las librerías propuesto el elegido es JQuery ya que ofrece las características necesarias para realizar el visor web. Además de ser el que se va a usar en el proyecto Primicia por los demás estudiantes.

¹³ Ajax: acrónimo de **A**synchronous **J**ava **S**cript **A**nd **X**ML (Java Script asíncrono y XML), es una técnica de desarrollo web para crear aplicaciones interactivas.

1.5 Conclusiones Parciales.

En el capítulo se analizaron los principales conceptos que sustentan el problema a resolver planteado, se realizó también un análisis del objeto de estudio, concluyendo que:

- El análisis de los principales conceptos asociados al dominio del problema permitió tener una mejor comprensión del sistema que se va a desarrollar.
- El análisis de las soluciones existentes en el ámbito nacional e internacional permitió estudiar las características de cada una de ellas y observar la forma en que esta visualizan las noticias. Llegando a la conclusión que estos sistemas no cumplen en su totalidad con las características necesarias para la plataforma PRIMICIA debido a que no muestran las noticias publicadas en la página principal, pero cumplen con características comunes que si se pueden utilizar a la hora de desarrollar este sistema.
- La caracterización de las tecnologías y herramientas propuestas para la elaboración de la aplicación permitió conocer las ventajas que ofrecen las mismas para el desarrollo del visor web.
- El uso de framework Symfony permite una construcción bien estructurada de la aplicación, optimizando el uso de lenguajes como JavaScript y PHP. El uso de RUP como metodología de desarrollo permite la generación de artefactos como los diagramas de clases del diseño, análisis, y los del modelo de implementación que son necesarios para una mayor comprensión de la aplicación a desarrollar. De esta manera se está en condiciones de realizar el desarrollo del módulo, utilizando las herramientas y tecnologías seleccionadas.

CAPÍTULO 2. VISOR WEB DE NOTICIAS DE LA PLATAFORMA DE TELEVISION INFORMATIVA PRIMICIA.

2.1 Introducción.

En el presente capítulo se realiza un estudio del sistema que se desea desarrollar mediante una descripción del modelo de dominio ya que no se cuenta con un negocio actual, así como la captura de los requisitos funcionales y no funcionales que debe tener el sistema. Además se realiza el diagrama de casos de uso del sistema, las descripciones de sus actores y casos de uso, concluyendo con el modelo de análisis y el modelo de diseño.

2.2 Modelo de Dominio

Modelo de Dominio o Conceptual: es un artefacto de la disciplina de análisis, es decir un diagrama, construido con las reglas de UML durante la fase de concepción con los objetos que existen (reales) relacionados con el proyecto que se va a acometer y las relaciones que hay entre ellos. Pero no son clases de software (aunque algunos objetos del Modelo de Dominio pueden terminar siéndolo). El mismo tiene como objetivo ayudar a comprender los conceptos que utilizan los usuarios. Es utilizado por el analista como un medio para comprender el sector de negocios al cual el sistema va a servir.(35) El proceso para su elaboración tiene tres pasos:

- Identificar las Clases Conceptuales.
- Dibujarlas en un Diagrama de Clases.
- Añadir Relaciones y Atributos.

2.2.1 Diagrama de clases del modelo de dominio.

En la figura 10 se muestran los principales conceptos asociados al dominio del problema y las relaciones que se establecen entre éstos.

CAPÍTULO II.PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA.

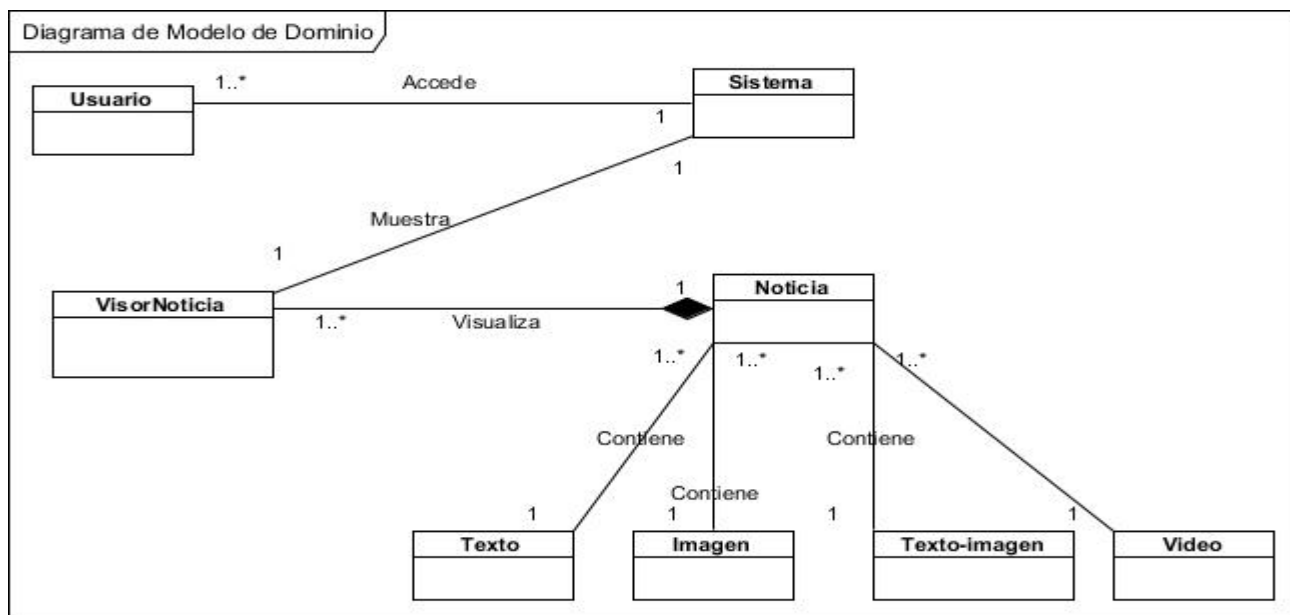


Figura 2: Diagrama de clase del dominio.

- **Usuario:** Persona encargada de interactuar con la aplicación.
- **Sistema:** Aplicación o la plataforma en la que el usuario interactúa.
- **Visor Noticia:** Interfaz web que permite visualizar las noticias.
- **Noticia:** Información que muestra el visor, las cuales pueden ser de tipo texto, texto-imagen, imagen y video.

2.3 Requisitos

Los requisitos de un sistema describen las funciones que ha de ofrecer el sistema y las restricciones asociadas a su funcionamiento. Por lo que es muy importante que se realice un buen levantamiento del mismo, para que el sistema quede con la calidad requerida. Los mismos deben cumplir determinadas características que son de gran importancia para que se entienda de una mejor forma.

2.3.1 Requisitos Funcionales

Un requisito funcional define las funciones que el sistema tiene que cumplir. Además describe las transformaciones que el sistema realiza sobre las entradas para producir salidas.

Los requisitos funcionales definidos para el desarrollo del visor web son los siguientes:

RF 1: Autenticar usuario: El sistema debe permitir al usuario autenticarse mediante su usuario y contraseña.

CAPÍTULO II. PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA.

RF2: Filtrar noticia. El sistema debe brindar la posibilidad de filtrar las noticias dados ciertos criterios de búsqueda (título, fecha o sección).

RF 3: Listar noticia. El sistema debe listar las noticias que se encuentren publicadas en la base de datos.

RF4: Seleccionar noticia. El sistema debe permitir seleccionar una noticia para ser visualizada.

RF 5: Visualizar noticia. El sistema debe permitir mostrar las noticias que estén publicadas cuando se acceda al módulo y cuando se seleccione en el listado.

RF 5.1: Pausar. El sistema debe permitir pausar la noticia que se esté mostrando.

RF 5.2: Anterior. El sistema debe mostrar la noticia que fue visualizada anteriormente.

RF 5.3: Siguiente. El sistema debe permitir mostrar la siguiente pantalla de la noticia o la próxima noticia.

RF 5.4: Reproducir la noticia de forma automática. El sistema debe permitir la reproducción de forma automática de las noticias.

2.3.2 Requisitos no funcionales

Los requisitos no funcionales tienen que ver con características que de una u otra forma puedan limitar el sistema, como por ejemplo, el rendimiento (en tiempo y espacio), interfaces de usuario, fiabilidad (robustez del sistema, disponibilidad de equipo), mantenimiento, seguridad, portabilidad y estándares. Dichos requisitos han de especificarse cuantitativamente siempre que sea posible (para que se pueda verificar su cumplimiento).

2.3.2.1 Software

Se debe utilizar Apache 2.0 como servidor web y PostgreSQL 9.0 o superior como sistema gestor de base de datos. Se deberá acceder al módulo a través de los navegadores más populares, como Mozilla Firefox 8.0 o superior, Chrome 16.0 o superior y cualquier versión de Internet Explorer igual o superior a la 8.0.

2.3.2.2 Hardware.

El servidor de Administración debe tener un procesador Pentium IV a 2.8 GHz o superior, con una memoria RAM de 1Gb como mínimo, un disco duro de 80Gb o mayor y una tarjeta de red Ethernet 10/100 Mbps o superior.

2.3.2.3 Apariencia

CAPÍTULO II. PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA.

La apariencia del módulo debe estar limitada por las mismas condiciones que PRIMICIA, haciendo uso de colores apropiados que denoten un producto profesional. La información debe presentarse de forma clara y organizada, permitiendo una correcta interpretación de los usuarios.

2.3.2.4 Usabilidad.

El subsistema podrá ser usado por personas con conocimientos básicos en el manejo de computadoras.

2.3.2.5 Portabilidad

El sistema estará realizado para funcionar en los sistemas operativos GNU/Linux o Windows, garantizando de esta manera un producto multiplataforma.

2.4 Definición de los actores del sistema

Como parte del sistema se identificó un actor para interactuar con el visor web.

Actor	Descripción
Usuario	Persona con acceso al sistema encargada de interactuar con las noticias que se muestran en el visor web.

Tabla 1 Actores Definidos del Sistema.

2.5 Diagrama de casos de uso del sistema (CUS).

En la figura 11 se muestra la relación del actor con los CUS identificados durante la realización del modelo de casos de uso del sistema.

CAPÍTULO II. PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA.

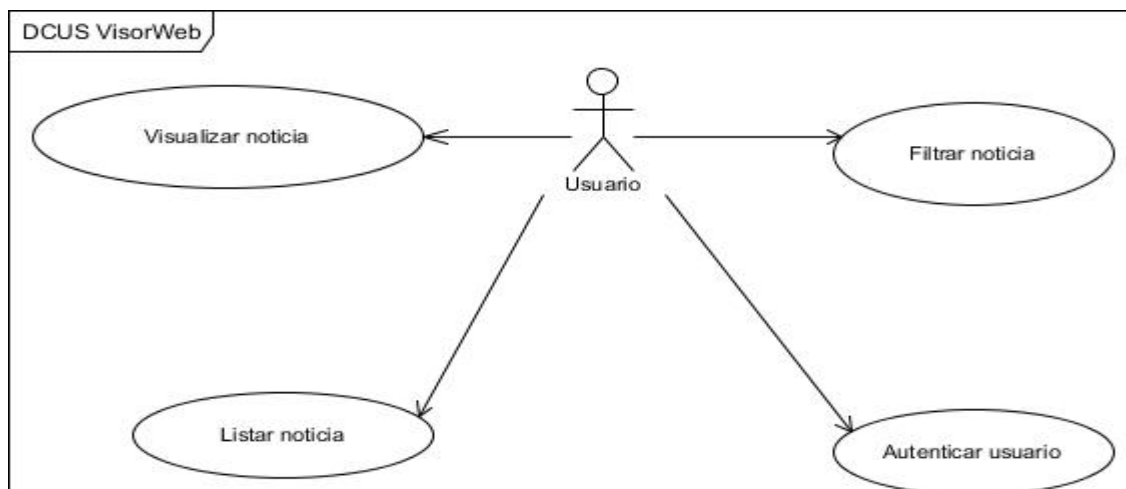


Figura 3: Diagrama de casos de uso del sistema.

2.6 Descripción textual de los casos de uso del sistema.

Caso de Uso:	Autenticar usuario
Actores:	Usuario
Resumen:	El caso de uso inicia cuando un usuario desea acceder al sistema, dando su usuario y contraseña, el caso de uso verifica la existencia de ese usuario, de no ser correcto, envía un mensaje informando que el usuario no es válido, en caso contrario lo lleva a la página de inicio según los privilegios de dicho usuario finalizando así el caso de uso.
Precondiciones:	El actor se encuentra en la página de autenticación del sistema, para ingresar en él.
Referencias	R1
Prioridad	Critico
Flujos Normal de Eventos	
Sección "Autenticar Usuario"	
Acción del Actor	Respuesta del Sistema

CAPÍTULO II. PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA.

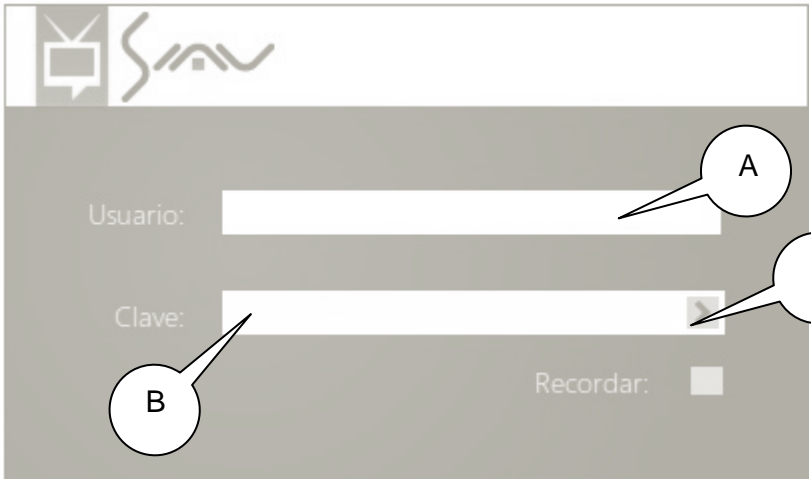
El usuario desea acceder al sistema.	Muestra un formulario (Interfaz 1) con dos campos para introducir el usuario (A) y la contraseña (B).
El actor ingresa los datos solicitados. Presiona el botón Aceptar (C).	Verifica que los datos ingresados por el usuario sean correctos. En ese caso aparecerá la página principal del sistema con las opciones que puede realizar este actor según su rol.
<p>Prototipo de Interfaz</p> <p>Interfaz 1</p> 	
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
Acción 3. El actor ingresa los datos solicitados. Presiona el botón Aceptar.	En caso que los datos no sean correctos aparecerá un mensaje informándole al usuario que no tiene acceso al sistema.
Poscondiciones	El usuario ha sido identificado y dados los privilegios según su configuración.

Tabla 2 Descripción del Caso de Uso Autenticar usuario.

CAPÍTULO II.PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA.

Caso de Uso:	Listar noticia
Actores:	Usuario
Resumen:	El caso de uso inicia cuando el actor accede al módulo visor. El sistema muestra un listado con las imágenes de la sección a la que pertenecen las noticias publicadas y el título de la misma. El caso de uso termina cuando se escoge una noticia del listado.
Precondiciones:	El actor accede al módulo visor.
Referencias	RF3
Prioridad	Crítico
Flujo Normal de Eventos	
Sección “Listar noticia”	
Acción del Actor	Respuesta del Sistema
1. El usuario accede al módulo Visor.	2. El sistema muestra un listado de las noticias que se encuentran publicadas en el sistema en forma de imagen con el logo de la sección a la que pertenece la misma y el título. Además tiene la posibilidad de visualizarla.(Interfaz 2)
Flujos Alternos	
1.1. El usuario accede al módulo Visor.	2.1 No hay noticas publicadas. El sistema muestra un mensaje “No hay resultados”.

CAPÍTULO II.PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA.


Prototipo de Interfaz Interfaz 2	
	
Poscondiciones	Se listan las noticias que están en transmisión en el sistema.

Tabla 3 Descripción del Caso de Uso Listar noticias.

Caso de Uso	Visualizar noticia
Actores:	Usuario
Resumen:	El caso de uso se inicia cuando se accede al módulo, donde el sistema muestra una interfaz con la reproducción automática de la noticia con el nombre de la sección, título de la noticia y el tipo de pantalla de la misma (texto, texto-imagen, imagen y video). Además se visualiza por parte del CU Listar noticias permitiendo poder seleccionar una noticia del listado para ser mostrada. También brinda las opciones de pausar, siguiente y anterior. El caso de uso termina cuando se muestra la noticia.
Precondiciones:	El usuario accede al módulo visor.
Referencias	RF3, RF4, RF5, RF5.1, RF5.2, RF5.3, RF5.4
Prioridad	Crítico
Flujo Normal de Eventos	
Sección “Visualizar noticia”	
Acción del Actor	Respuesta del Sistema
1. El usuario accede al módulo Visor.	2. El sistema carga en la interfaz una noticia con la información general (sección a la que está relacionada, título y su contenido) (Interfaz 3). Se muestra además las opciones de Pausar (D), Anterior (E) y Siguiente (F).

CAPÍTULO II.PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA.

<p>3. El usuario selecciona la noticia que desea que se muestra.</p>	<p>4. El sistema muestra la noticia con la información general de la noticia (sección a la que está relacionada, título y su contenido).Se muestra además las opciones de Pausar, Anterior, Siguiente.</p>
<p>5. Selecciona una opción de las que brinda el sistema.</p>	<p>6. El sistema realiza la operación según la opción que escoja el usuario:</p> <p>a) Si selecciona Pausar, ver sección Pausar.</p> <p>b) Si selecciona Anterior, ver sección Anterior.</p> <p>c) Si selecciona Siguiente, ver sección Siguiente.</p>
Sección "Pausar "	
<p>1. El usuario presiona el botón Pausar de la noticia que se está visualizando.</p>	<p>2. El sistema detiene la reproducción de la noticia.</p>
Sección "Anterior"	
<p>1. El usuario presiona el botón anterior de la noticia que se está visualizando.</p>	<p>2. El sistema muestra la noticia anterior que fue visualizada.</p>
Sección "Siguiente"	
<p>1.El usuario presiona el botón siguiente de la noticia que se está visualizando.</p>	<p>2.El sistema muestra la siguiente pantalla de la noticia o la siguiente noticia.</p>
<p>Prototipo de Interfaz Interfaz 3</p>	

CAPÍTULO II.PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA.



Flujos Alternos	
Acción del Actor	Respuesta del Sistema
1.1 El usuario accede al módulo visor y no exista ninguna noticia publicada en la base de datos.	2.1 El sistema muestra la interfaz vacía.
Poscondiciones	Se mostró la noticia seleccionada.

Tabla 4 Descripción del Caso de Uso Visualizar noticias.

Caso de Uso:	Filtrar noticia
Actores:	Usuario

CAPÍTULO II.PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA.

Resumen:	El caso de uso inicia cuando el actor presiona el botón filtrar .El sistema muestra una formulario para que el usuario introduzca los criterios por lo que desea filtrar. Luego se presiona el botón Filtrar y el sistema muestra un listado con las noticias que coinciden con los criterios de búsqueda introducidos. El caso de uso termina cuando el usuario realiza la búsqueda.	
Precondiciones:	El accede al módulo visor.	
Referencias	RF2	
Prioridad	Crítico	
Flujos Normal de Eventos		
Sección “Filtrar noticia ”		
Acción del Actor	Respuesta del Sistema	
1. El usuario accede al módulo visor.	2. El sistema muestra en la interfaz principal el botón Filtrar (G) y Recargar (H).	
3. El usuario presiona el botón Filtrar (G).	4. El sistema muestra un formulario con un campo de texto (I) para introducir el título, el rango de fecha (J) y una lista para seleccionar la sección temática (K) a la que pertenece la noticia. (Interfaz 4). Además muestra los botones Filtrar (L) y Cancelar (M).	
5. El actor introduce algún criterio de búsqueda y presiona el botón Filtrar (L).	6. Actualiza el listado con las noticias que concuerden con el criterio de búsqueda introducido.	
Flujos Alternos		

CAPÍTULO II. PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA.

5.1 El usuario no introduce ningún criterio de búsqueda y presiona el botón Filtrar (L).	5.2 El sistema muestra todas las noticias publicadas.
5.3 El usuario introduce datos incorrectos y presiona el botón Filtrar (L).	5.4 El sistema muestra un mensaje "No hay resultados".
5.5 El usuario presiona el botón Recargar (H).	5.6 El sistema muestra el listado de todas las noticias.

Prototipo de Interfaz

Interfaz 4

El prototipo de interfaz muestra un formulario con los siguientes elementos:

- G**: Icono de lista.
- H**: Botón de recargar.
- I**: Campo de texto para el título.
- J**: Selector de sección.
- K**: Campos de fecha para el inicio y fin.
- L**: Botón de filtrar.
- M**: Botón de cancelar.

Poscondiciones	Se filtran las noticias según el criterio introducido por el usuario.
-----------------------	---

Tabla 5 Descripción del Caso de Uso Filtrar noticias.

2.7 Modelo de análisis

Luego que se identifican los requisitos y se describen los determinados casos de uso del sistema es necesario realizar un análisis de estos, para así tener una mejor comprensión antes de comenzar a diseñar. Es por lo que el modelo de análisis se puede considerar como una primera aproximación al modelo de diseño. En dicho análisis se sigue un paradigma orientado a objetos (OO) por lo que se obtienen las primeras propuestas de clases que se encarguen de garantizar los distintos servicios que el cliente necesita. Dicha disciplina tiene como propósito:

CAPÍTULO II. PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA.

- Conseguir una comprensión más precisa de los requisitos, refinarlos y estructurarlos.
- Utilizar el lenguaje de los desarrolladores para analizar con profundidad los requisitos funcionales.
- Proporcionar una visión general del sistema.

En el análisis se modelan solo 3 tipos de clases (interfaz, control, entidad)

2.7.1 Diagramas de clases del análisis

En los siguientes diagramas se representan las clases del análisis y las relaciones que se establecen entre cada una de ellas.

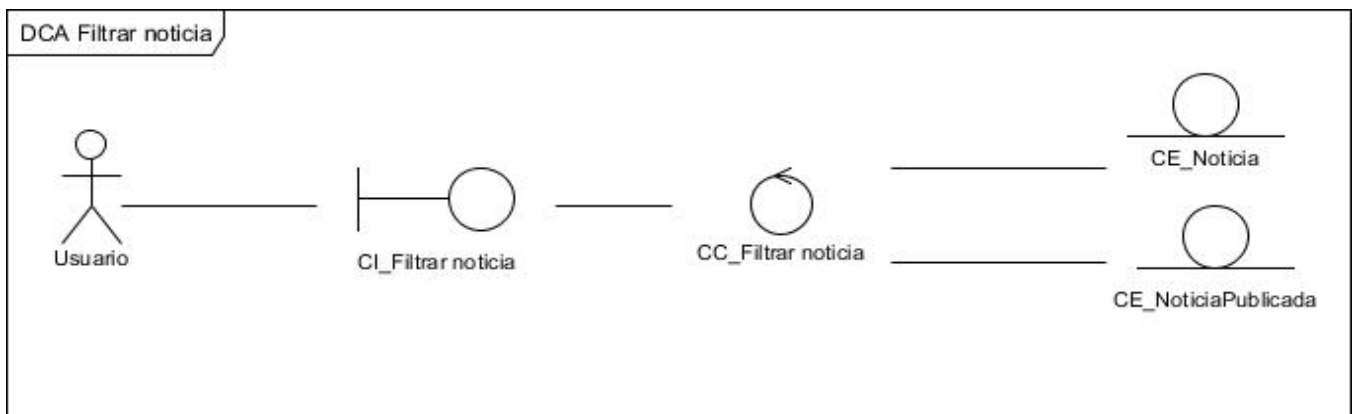


Figura 4: Diagrama de clases del análisis del CUS: Filtrar noticia.

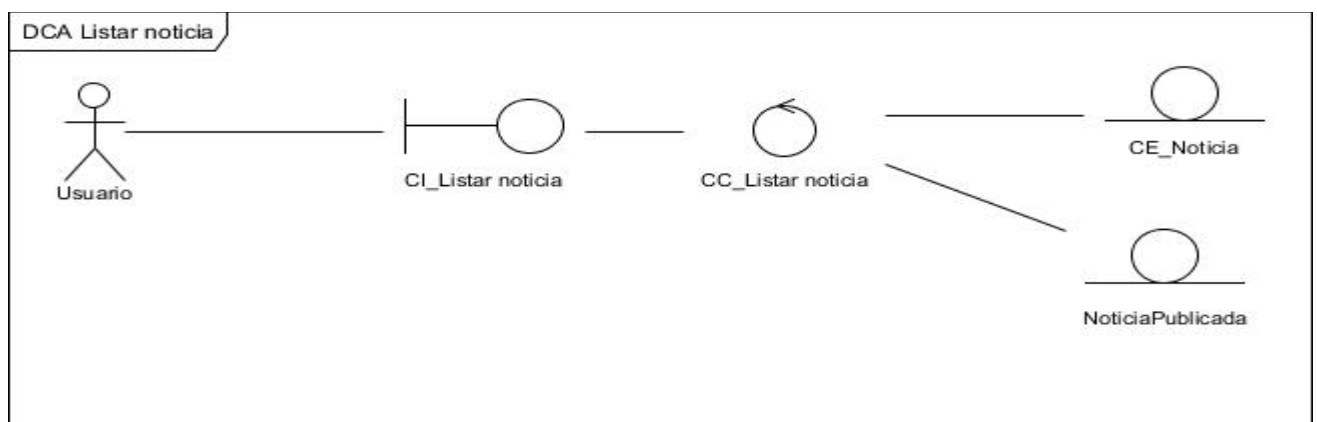


Figura 5: Diagrama de clases del análisis del CUS: Listar noticia.

CAPÍTULO II.PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA.

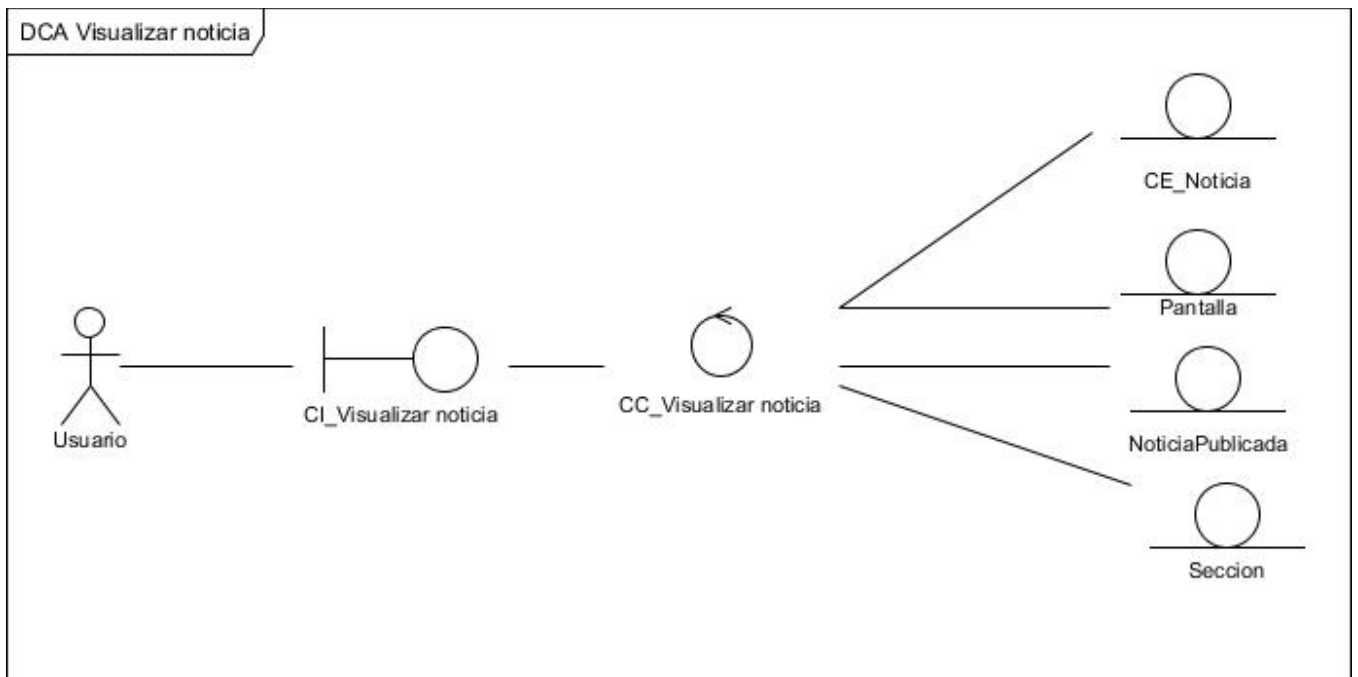


Figura 6: Diagrama de clases del análisis del CUS: Visualizar noticia.

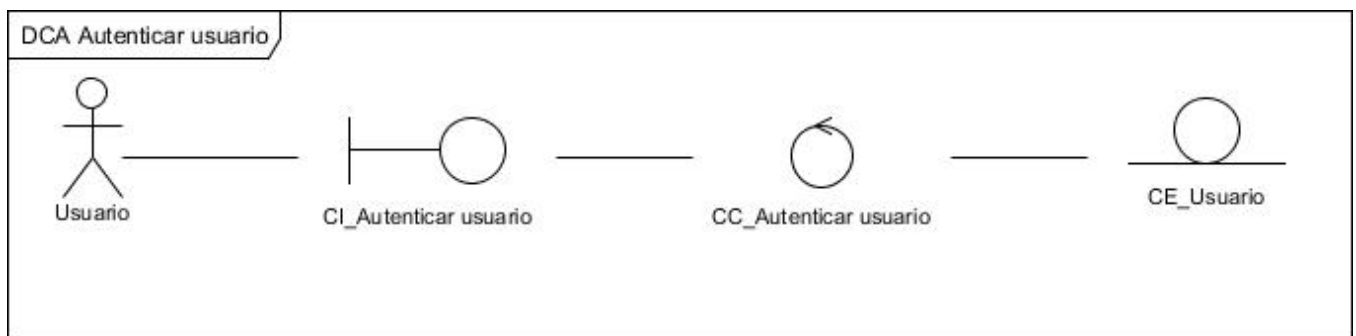


Figura 7: Diagrama de clases del análisis del CUS: Autenticar usuario.

2.8 Modelo de diseño

El modelo de diseño es un modelo de objetos que describe la relación física de los casos de uso centrándose en como los requisitos funcionales y no funcionales, junto con otras restricciones relacionadas con el entorno de implementación, tienen impacto en el sistema a considerar. Además, el modelo de diseño sirve de abstracción de la implementación del sistema y es, de ese modo, utilizada como una entrada fundamental de las actividades de implementación.

2.8.1 Arquitectura MVC

El uso del framework que utiliza MVC obliga a dividir y organizar el código de acuerdo a las convenciones establecidas por el framework. Dicha arquitectura está formada por 3 niveles, que son independientes y la comunicación entre ellos es mediante interfaces que abstraen sus estructuras internas.

El código de la presentación se guarda en la vista, el código de manipulación de datos se guarda en el modelo y la lógica de procesamiento de las peticiones constituye el controlador.

Symfony toma lo mejor de la arquitectura MVC y la realiza de modo que el desarrollo de aplicaciones sea rápido y sencillo.

- En el controlador se encuentran las acciones, las cuales son el núcleo de la aplicación, pues contienen toda la lógica de la aplicación. Estas acciones utilizan el modelo y precisan las variables para la vista. Al realizarse una petición web en una aplicación Symfony, la URL define una acción y los parámetros de la petición. En Symfony2 todas las peticiones se realizan a través del controlador frontal `app.php`, el cual a través del enrutamiento delega las responsabilidades en las clases Controller de cada Bundle.
- La vista es la encargada de originar las páginas que son mostradas como resultado de las acciones, donde se encuentra el *layout*, que es común para todas las páginas de la aplicación. En Symfony2 la capa de la vista está formada por plantillas *twig*. Estas plantillas se guardan en directorios llamados *views/* dentro de cada bundle.
- En el Modelo se encuentran las clases, que son generadas de forma automática según la estructura de la BD.Symfony2 guarda todas las clases y archivos relacionados con el modelo en el directorio `src/VisorBundle/Entity`, estas clase son generadas por las librerías de Doctrine2. Doctrine es el motor generador que se encarga de esta generación automática para construir sus clases, creando la estructura y generando el código de las mismas. A medida que el desarrollo de un proyecto va avanzando, puede ser necesario agregar métodos y propiedades personalizadas en los objetos del modelo, esto trae consigo que se aumenten las tablas o columnas. Asimismo, cada vez que se modifica se deben regenerar las clases del modelo de objetos. Si se añaden los métodos personalizados en las clases que se generan, cada vez que se vuelvan a generar esas clases estos métodos se borrarían.

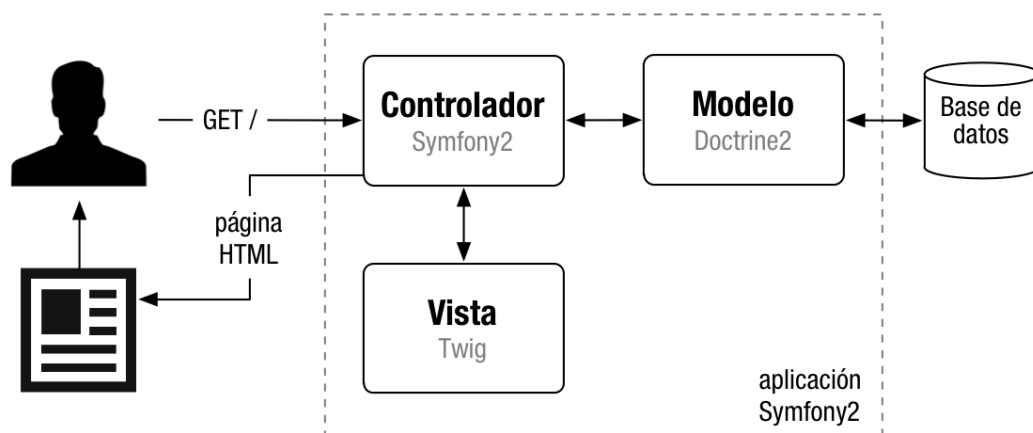


Figura 8: Esquema simplificado del funcionamiento interno de Symfony2.

2.8.2 Patrones de diseño

A la hora de desarrollar un Software es importante el uso de Patrones de diseño, pues “son el esqueleto de las soluciones a problemas comunes en el desarrollo de software.” En otras palabras, brindan una solución ya probada y documentada a problemas de desarrollo de software, constituyen una guía para resolver problemas comunes en programación. Su utilización ayuda a obtener un software de calidad, que sea reutilizable y que permita ampliar sus funcionalidades.

Los patrones de diseño son muchos y muy variados, resulta casi imposible revisar todos los que existen a la hora de desarrollar una aplicación. Por este motivo se recomienda el uso de los patrones que estén predeterminados en cada uno de los estilos que se seleccionen para la arquitectura que este caso será Symfony2.

Entre los patrones de diseño que implementa se encuentran:

Patrones GRASP¹⁴

Experto: Es uno de los patrones que más se utiliza cuando se trabaja con Symfony, con la inclusión de la librería Doctrine para mapear la Base de Datos. Symfony utiliza esta librería para realizar su capa de abstracción en el modelo, encapsular toda la lógica de los datos y generar las clases con todas las funcionalidades comunes de las entidades.(36)

¹⁴ GRASP es un acrónimo que significa General Responsibility Assignment Software Patterns (Patrones generales de software para asignar responsabilidades)

CAPÍTULO II.PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA.

Creador: Guía la asignación de responsabilidades relacionadas con la creación de objetos, tarea muy frecuente en los sistemas orientados a objetos. El propósito fundamental de este patrón es encontrar un creador para una determinada clase.

En la clase `DefaultController` se encuentran las acciones definidas para el módulo visor web y se ejecutan cada una de ellas. En las acciones se crean los objetos de las clases que representan las entidades, evidenciando de este modo que la clase `DefaultController` es la creadora de dichas entidades.(36)

Bajo Acoplamiento: Estimula asignar una responsabilidad de modo que no incremente el acoplamiento al punto de producir resultados negativos propios de un alto acoplamiento.

El Bajo Acoplamiento soporta el diseño de clases más independientes, que reducen el impacto de los cambios, y también más reutilizables, que acrecientan la oportunidad de una mayor productividad. No puede considerarse en forma independiente de otros patrones como Experto o Alta Cohesión, sino que más bien ha de incluirse como uno de los principios del diseño que influyen en la decisión de asignar responsabilidades.

En `Symfony2`, la clase `DefaultController` hereda solamente de `Controller`, que es una clase estable en cuanto a su implementación, garantizando un bajo acoplamiento entre las clases.(36)

Alta Cohesión es un patrón evaluativo que el desarrollador aplica al valorar sus decisiones de diseño. Se da una alta cohesión funcional cuando los elementos de un componente (clase, por ejemplo) colaboran para producir algún comportamiento bien definido.

`Symfony2` permite asignar responsabilidades a una clase con una alta cohesión pues plantea que debe existir una acción para cada una de las plantillas que producen la vista. Además la clase `Controller` colabora con otras para realizar diferentes operaciones, instanciar objetos y acceder a las propiedades, proporcionando que el software sea flexible frente a grandes cambios.(36)

Controlador: es un patrón que sirve como intermediario entre una interfaz y el algoritmo que la implementa, de tal forma es la que recibe los datos del usuario y los envía a las distintas clases según el método llamado. Este patrón sugiere que la lógica de negocios debe estar separada de la capa de presentación, esto para aumentar la reutilización de código y a la vez tener un mayor control. En `Symfony2` el controlador *recibe* la petición (`Request`) y se encarga de crear y devolver una respuesta (`Response`)".Este patrón se evidencia en las clases `DefaultController`. (36)

Patrones GoF¹⁵

Patrones Creacionales:

Abstract Factory: También conocido como Fábrica abstracta, permite trabajar con objetos de distintas familias de manera que las familias no se mezclen entre sí haciendo transparente el tipo de familia concreta que se esté usando.

Cuando el framework necesita por ejemplo crear un nuevo objeto para una petición, busca en la definición de la factoría el nombre de la clase que se debe utilizar para esta tarea. Como la definición por defecto de la factoría para las peticiones es `getRequest()`, Symfony2 crea un objeto de esta clase para tratar con las peticiones.(36)

Patrones Estructurales:

Decorator: También conocido como Decorador añade funcionalidad a una clase de una forma dinámica y transparente.

En Symfony2 los archivos `base.html.twig` y `layout.html.twig` almacenan el código HTML que es común a todas las páginas de la aplicación, para no tener que repetirlo en cada página. El contenido de las plantillas se integra en estos archivos, o si se mira desde el otro punto de vista, el archivo base y el layout decoran el contenido de las plantillas.(37)

2.8.3 Diagramas de clases del diseño

Un diagrama de clases es un diagrama estático que describe la estructura de un sistema mostrando sus clases, atributos y las relaciones entre ellos. Los diagramas de clases son utilizados durante el proceso de análisis y diseño de los sistemas para lograr una mejor interpretación de la implementación del sistema. Estos diagramas describen de las clases sus atributos, métodos y las relaciones entre ellas.

A continuación, se muestran los diagramas de clases del diseño de los casos de usos más importantes representados siguiendo la arquitectura definida.

Modelo, Vista, y Controlador: representan paquetes lógicos en correspondencia con la arquitectura del framework.

Doctrine: representa de manera lógica el ORM seleccionado para implementación del nuevo sistema y responde al deseo del proyecto PRIMICIA de utilizar las versiones más actuales del framework.

¹⁵GoF (Gang of Four) libro Design Patterns escrito por el grupo compuesto por Erich Gamma, Richard Helm, Ralph Johnson y John Vlissides.

CAPÍTULO II. PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA.

Framework Symfony: paquete estereotipado <<framework>> representa las clases y elementos que permiten su funcionamiento.

Librería JQuery: paquete estereotipado que representa todos archivos JavaScript que permiten el funcionamiento del sistema.

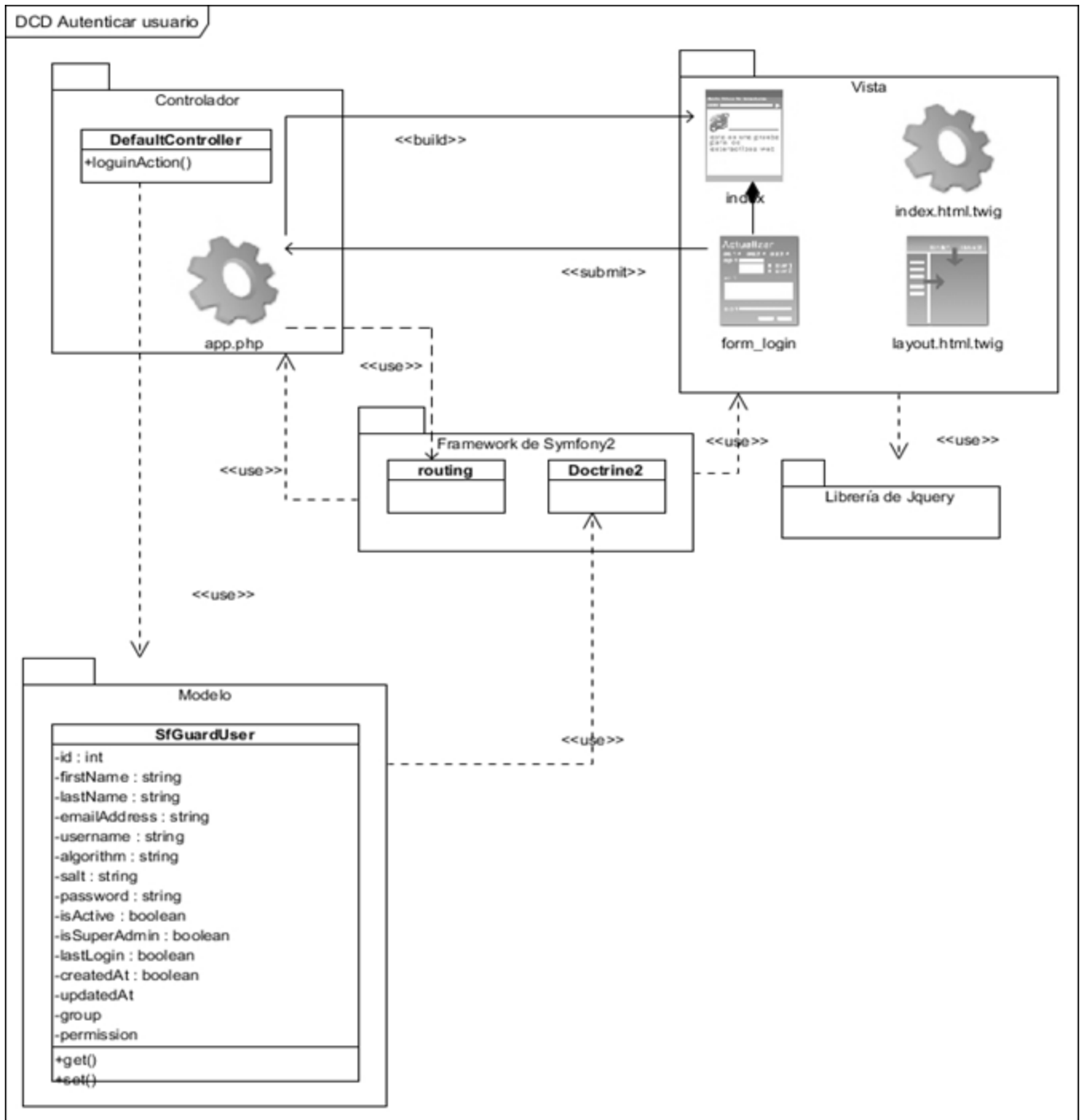


Figura 9: Diagrama de clase de diseño. CU Autenticar usuario

CAPÍTULO II.PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA.

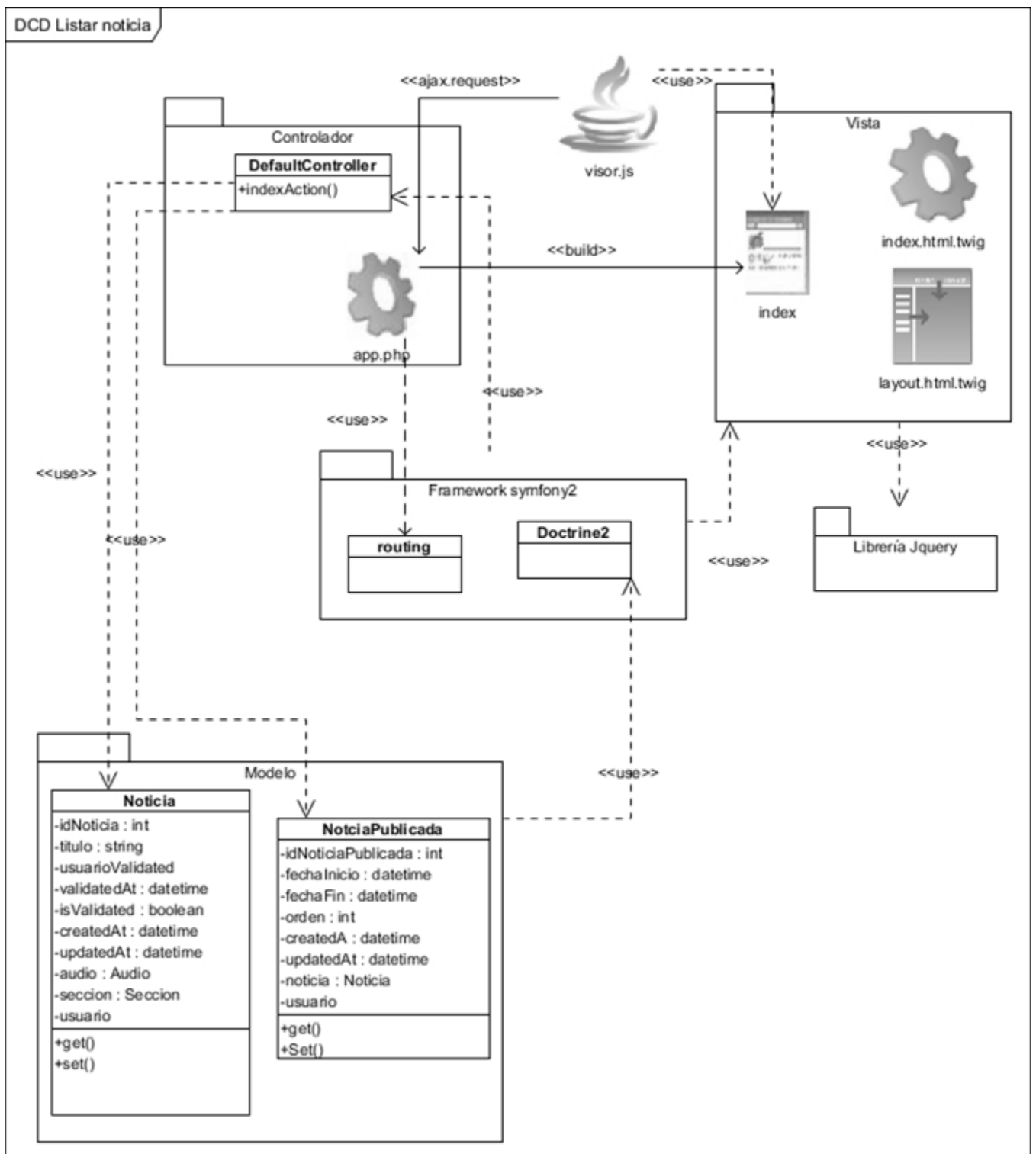


Figura 10: Diagrama de clase de diseño. CU Listar noticia

CAPÍTULO II.PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA.

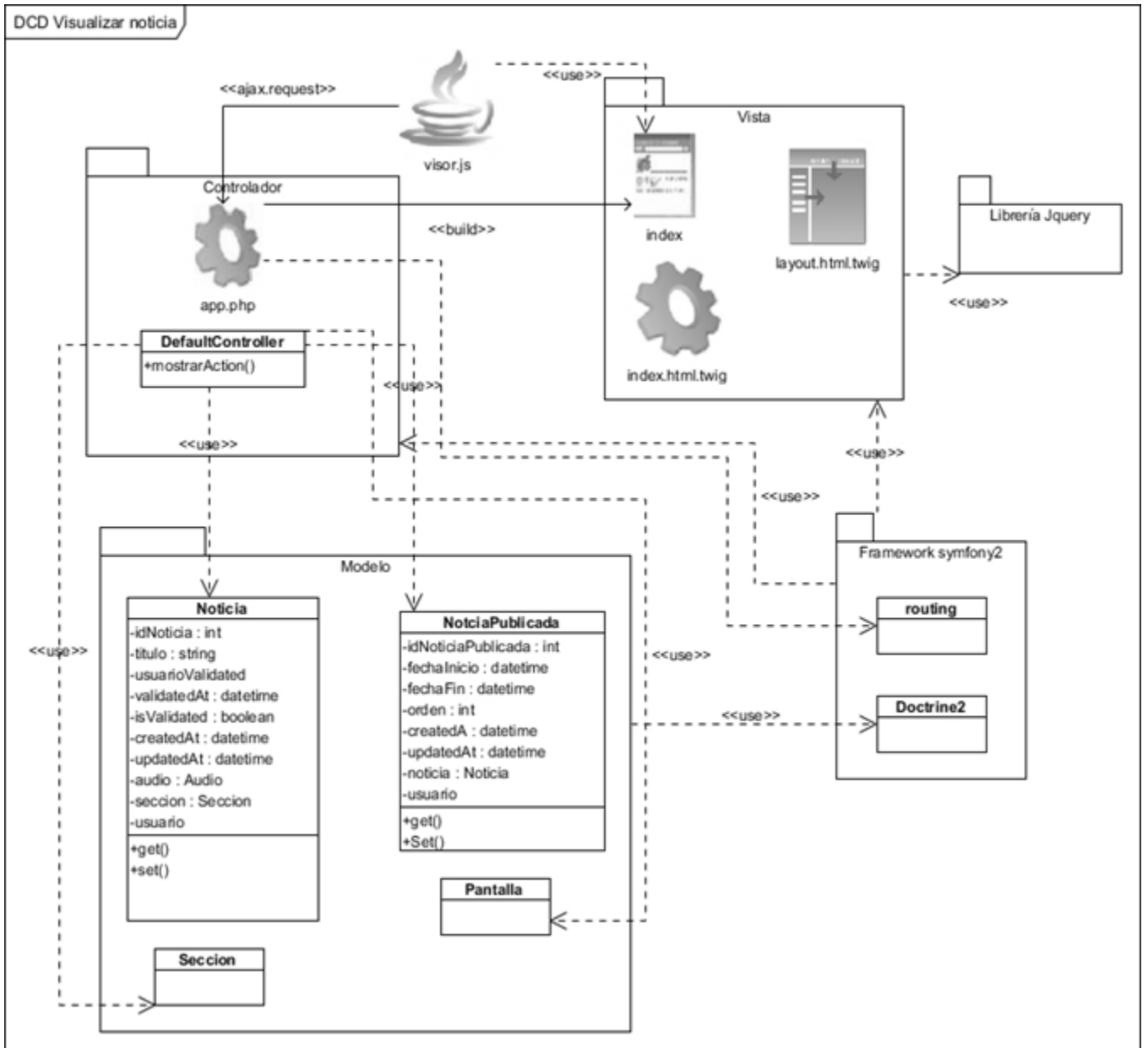


Figura 11: Diagrama de clase de diseño. CU Visualizar noticia

CAPÍTULO II.PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA.

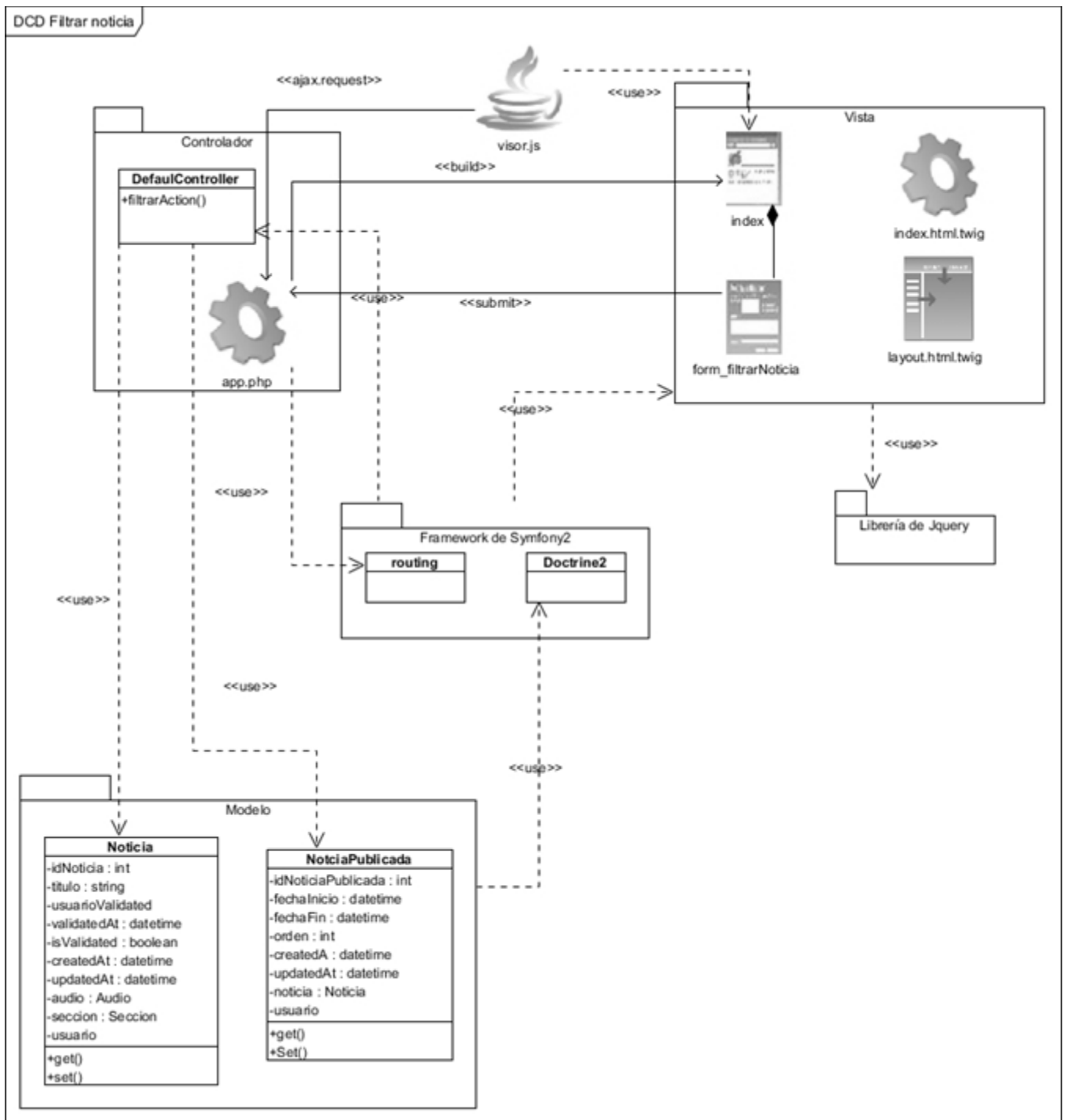


Figura 12: Diagrama de clase de diseño. CU Filtrar noticia

2.9 Conclusiones Parciales.

CAPÍTULO II.PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA.

En este capítulo se realizó una descripción de la solución propuesta a implementar. En el mismo se analizaron los principales aspectos que están relacionados con el proceso de desarrollo del visor web:

- Se describieron los requisitos funcionales y no funcionales del sistema, los cuales representan las principales funcionalidades a implementar.
- Se realizó una descripción de los casos de uso, los cuales muestran una guía de cómo va a estar constituido el visor web con todas sus funcionalidades.
- El uso de patrones permitió elaborar diagramas de clases de diseño fiables para la implementación de los diferentes casos de uso del visor web guiado por la arquitectura Modelo-Vista-Controlador.
- La ejecución de los flujos de trabajo establecidos en la metodología seleccionada permitió el desarrollo del sistema en el tiempo establecido.

CAPÍTULO III.VALIDACIÓN DE LA SOLUCIÓN PROPUESTA.

CAPÍTULO 3 VALIDACIÓN DEL VISOR WEB DE NOTICIAS DE LA PLATAFORMA DE TELEVISIÓN INFORMATIVA PRIMICIA.

3.1 Introducción.

En este capítulo se muestra el resultado de la implementación y prueba de la solución propuesta. En el mismo se mostrará los diagramas del modelo de implementación así como, el modelo de despliegue. Además de las pruebas que se le realizaron a la aplicación.

3.2 Modelo de Implementación

El Modelo de Implementación es comprendido por un conjunto de componentes y subsistemas que constituyen la composición física de la implementación del sistema. Entre los componentes podemos encontrar datos, archivos, ejecutables, código fuente y los directorios. Fundamentalmente se describe la relación que existe desde los paquetes y clases del modelo de diseño a subsistemas y componentes físicos. Este artefacto describe cómo se implementan los componentes, asociándolos en subsistemas organizados en capas y jerarquías. (38)

3.2.1 Diagramas de componentes

El diagrama de componentes describe la relación entre componentes de software, sus dependencias, su comunicación y su ubicación. Muestran las opciones de realización incluyendo código fuente, binario y ejecutable. Los componentes representan todos los tipos de elementos de software que entran en la fabricación de aplicaciones informáticas. Pueden ser simples archivos, paquetes y bibliotecas cargadas dinámicamente.(39)

En las figuras siguientes se muestran los diagramas de componentes de los casos de uso reflejando cada archivo en cada uno de los paquetes, guiado por la arquitectura definida Modelo Vista Controlador.

CAPÍTULO III. VALIDACIÓN DE LA SOLUCIÓN PROPUESTA.

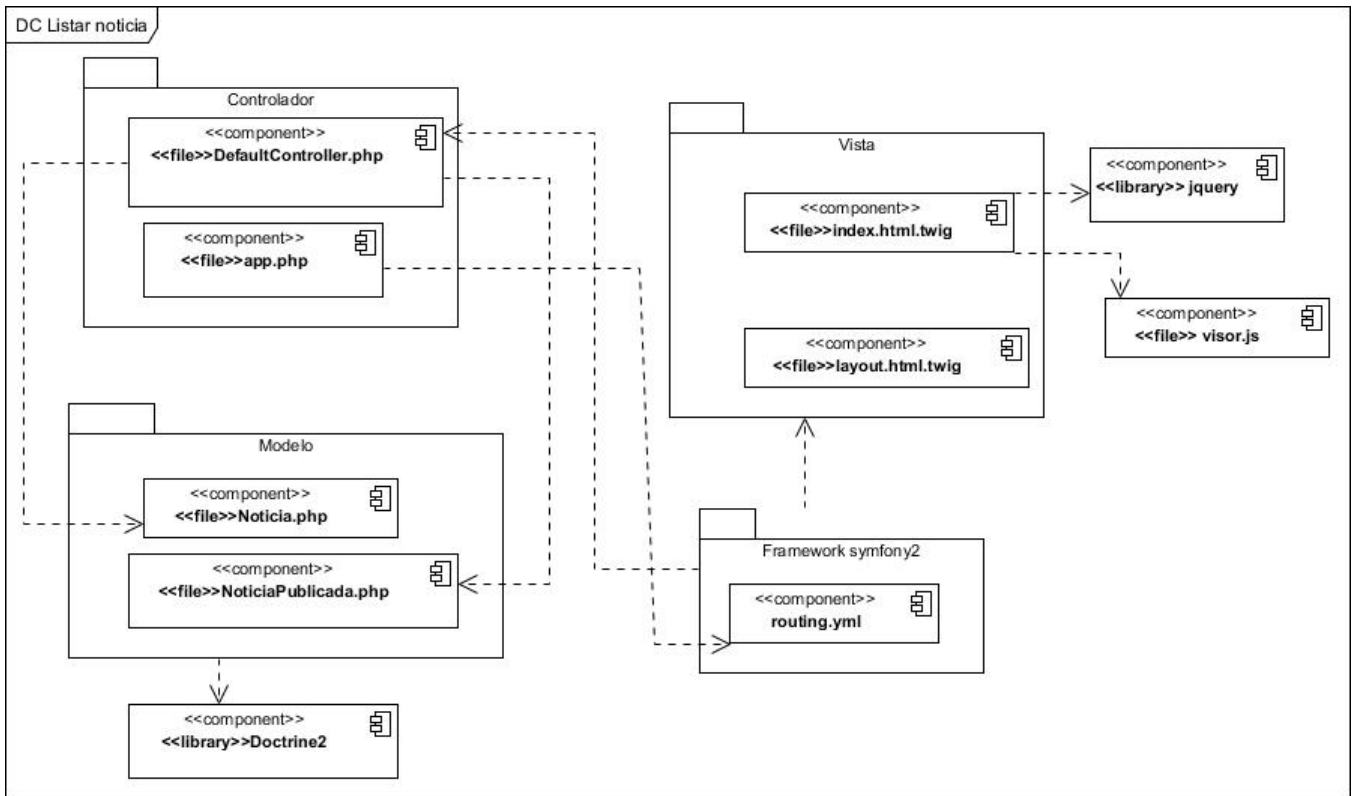


Figura 13 Diagrama componente. CU Listar noticia

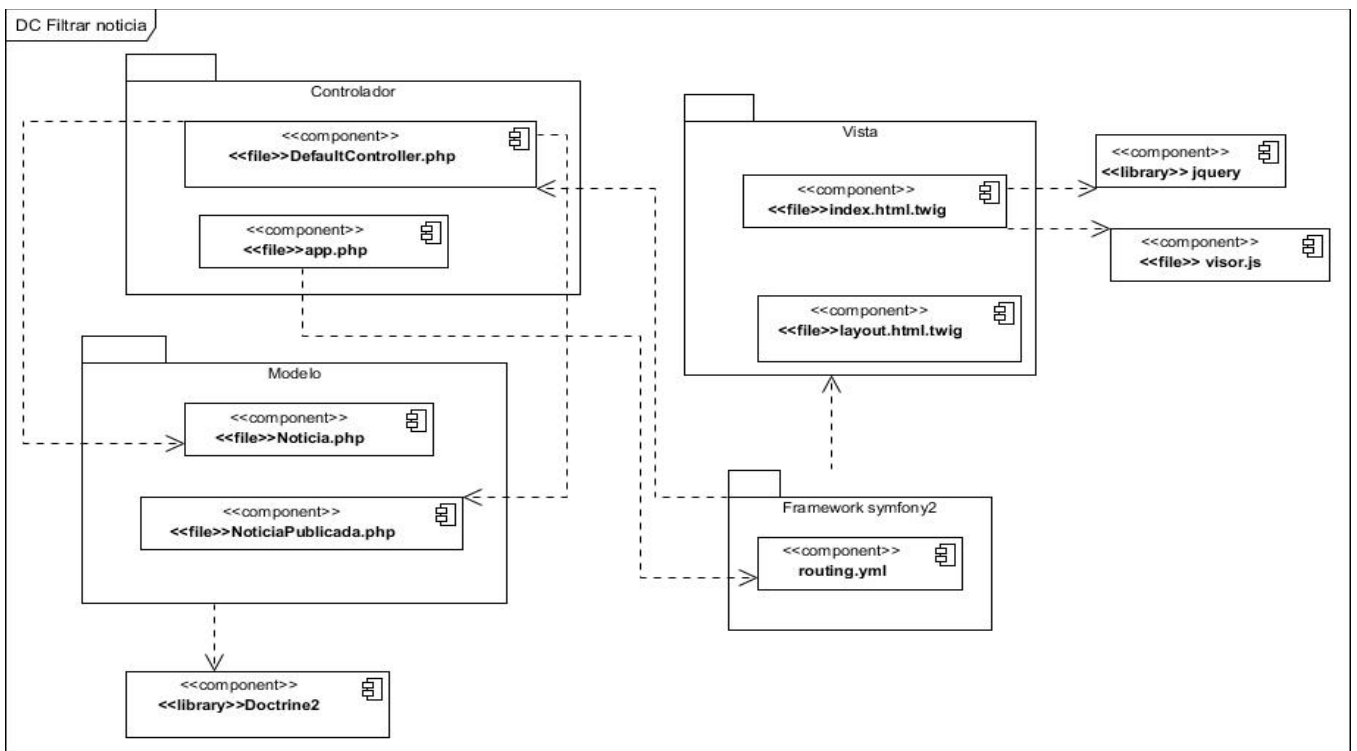


Figura 14 Diagrama componente. CU Filtrar noticia

CAPÍTULO III. VALIDACIÓN DE LA SOLUCIÓN PROPUESTA.

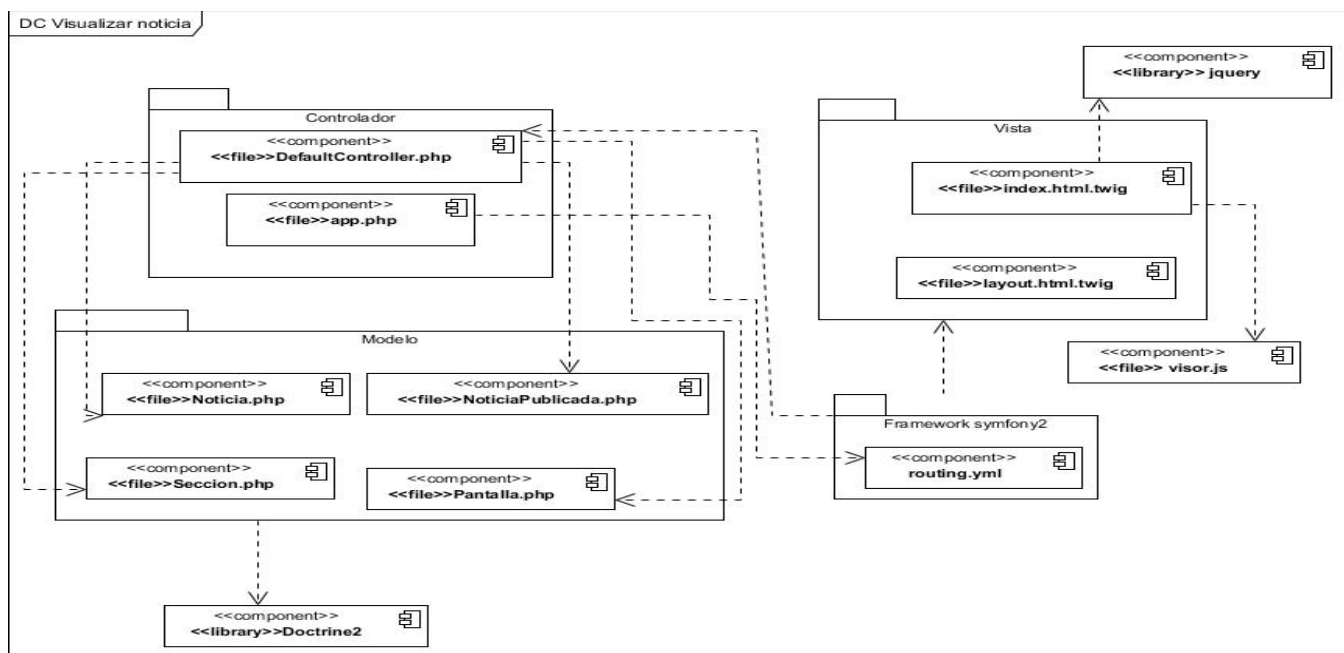


Figura 15 Diagrama componente. CU Visualizar noticia

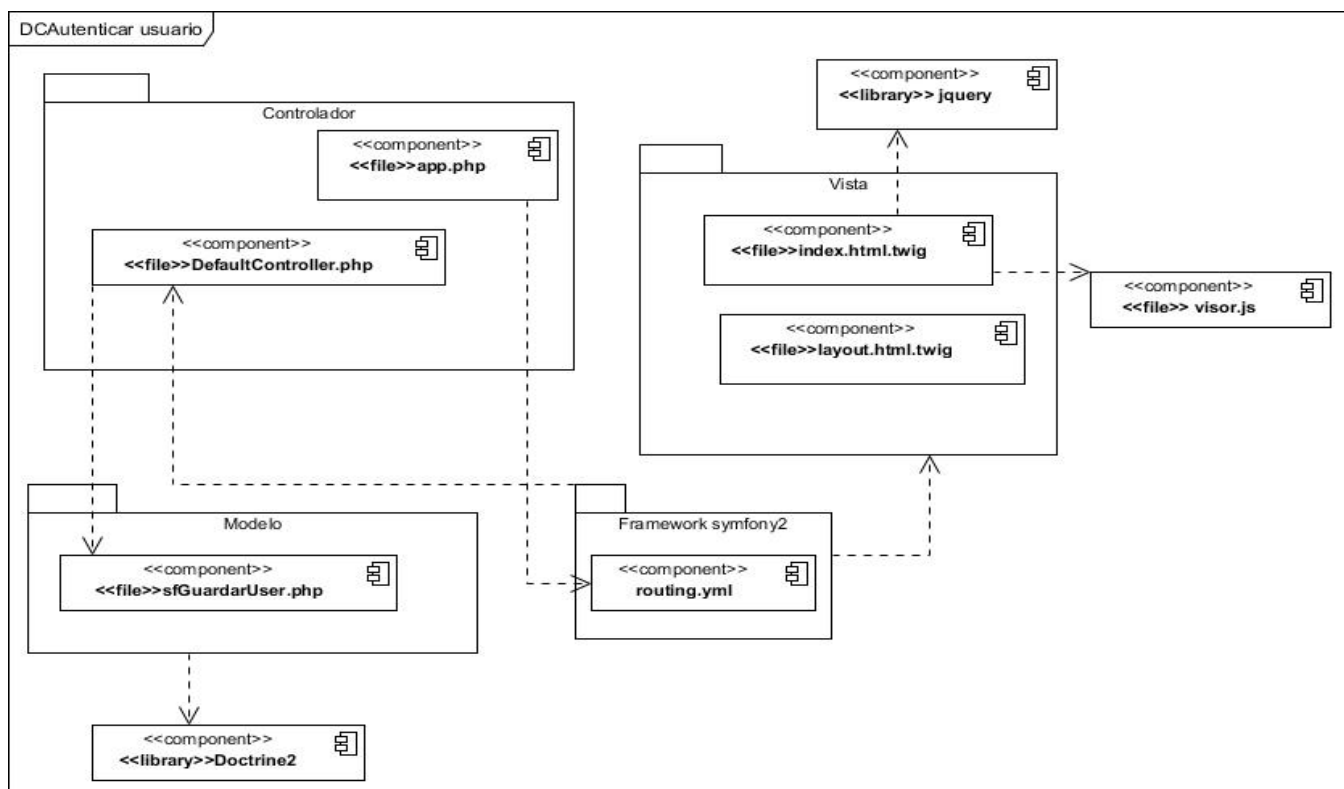


Figura 16 Diagrama componente. CU Autenticar usuario

3.3 Modelo de Despliegue

El modelo de despliegue es un modelo de objetos que describe la distribución física del sistema en términos de cómo se distribuye la funcionalidad entre los nodos de cómputo. Cada nodo representa un recurso de cómputo, normalmente un procesador o un dispositivo hardware similar. Puede describir diferentes configuraciones de red, incluidas las configuraciones para prueba y para simulación. (38)

3.3.1 Diagramas de despliegue

Descripción de los nodos

PC cliente: nodo cliente mediante el cual el usuario usando el navegador web Mozilla Firefox versión 3.6 o superior accede a la aplicación a través del protocolo http.

Nodo Servidor de Administración: Este nodo representa el servidor donde estará alojado el Subsistema de Administración, el cual tendrá como sistema gestor de base de datos, PostgreSQL 9.0 y como servidor web, Apache2.

Protocolo HTTP: Este protocolo se utiliza para el acceso a sitios web. Consta de cuatro etapas entre el navegador y el servidor, conexión, solicitud, respuesta y desconexión y por lo general trabaja sobre el puerto 80.

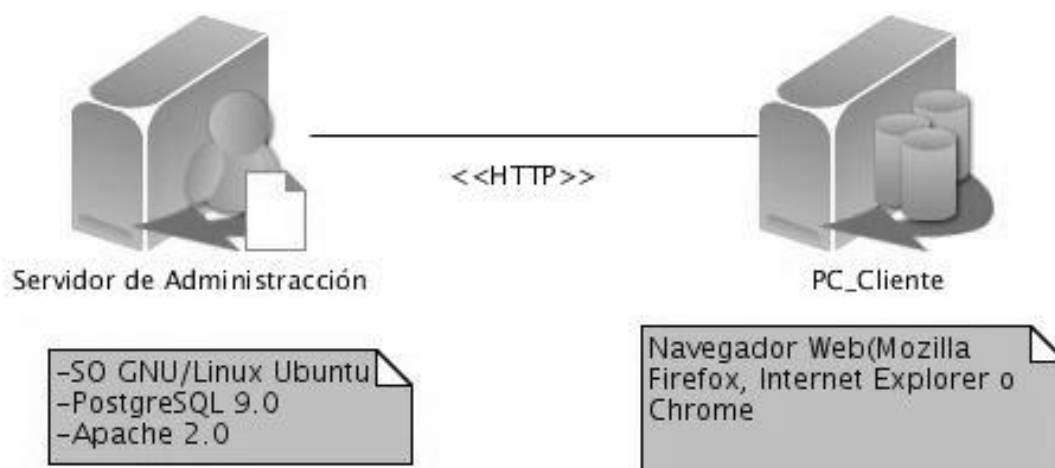


Figura 17 Modelo de despliegue

3.4 Pruebas

Algo que es de vital importancia durante el desarrollo de software es la realización de pruebas. Las pruebas de software es el proceso orientado a demostrar que un programa no tiene errores, que realiza las funciones para las cuales fue construido.

CAPÍTULO III. VALIDACIÓN DE LA SOLUCIÓN PROPUESTA.

En este proceso se ejecutan pruebas dirigidas a componentes del software o al sistema de software en su totalidad, con el objetivo de medir el grado en que el software cumple con los requisitos. En las pruebas se usan casos de prueba, especificados de forma estructurada mediante Técnicas de prueba. Existen dos métodos para la realización de las mismas, las pruebas de caja blanca o estructural y las pruebas de caja negra o funcional.

3.4.1 Pruebas de caja negra.

La prueba de caja negra se refiere a las pruebas que se llevan a cabo sobre la interfaz del software. Están destinadas para validar los requisitos funcionales sin fijarse en el funcionamiento interno de un programa.

Entre sus funciones principales están:

- Verifican las especificaciones funcionales y no consideran la estructura interna del programa.
- Es hecha sin el conocimiento interno del producto.
- No validan funciones ocultas (por ejemplo funciones implementadas pero no descritas en las especificaciones funcionales del diseño) por tanto los errores asociados a ellas no serán encontrados.

En otras palabras, la prueba de la caja negra se refiere a las pruebas que se llevan a cabo sobre la interfaz del software. O sea los casos de prueba pretenden:

- Demostrar las funciones del software son operativas.
- Que las entradas se aceptan de la forma adecuada y que se produce el resultado correcto.

3.4.1.1 Técnica de Partición Equivalente

Al módulo implementado se le realizará esta prueba y particularmente la técnica de prueba partición equivalente el cual divide el campo de entrada en clases de datos que tienden a ejercitar determinadas funciones del software. La partición equivalente se esfuerza por definir un caso de prueba que descubra varias clases de errores, reduciendo así el número de casos de prueba que deben desarrollarse. Se definen un conjunto de estados válidos y no válidos para las condiciones de entrada y se verifica que los datos de salida sean los esperados.

Se realizaron 3 diseños de Caso de Prueba (CP) que a continuación se muestran:

CAPÍTULO III. VALIDACIÓN DE LA SOLUCIÓN PROPUESTA.

Nombre de la sección	Escenarios de la sección	Descripción de la funcionalidad	Respuesta del sistema	Resultado de la prueba	Flujo Central
SC 1: Filtrar noticia	EC 1.1: Filtrar noticias con éxito.	El usuario selecciona el botón Filtrar. Luego el sistema muestra un formulario con los campos por lo se desea filtrar (título, sección, fecha inicio y fecha fin de publicación de la noticia). Se llenan los datos correspondientes. Se presiona el botón filtrar. El sistema muestra las noticias que coinciden con el filtro.	El sistema actualiza el listado de noticia que se encuentra en la interfaz principal con los resultados del filtro.	Satisfactorio.	Visor/ Clic en el botón Filtrar.

CAPÍTULO III. VALIDACIÓN DE LA SOLUCIÓN PROPUESTA.

	EC 1.2: Filtrar noticias éxito.	El usuario selecciona el botón Filtrar. Luego el sistema muestra un formulario con los campos por lo se desea filtrar (título, sección, fecha inicio y fecha fin de publicación de la noticia). Se dejan los campos vacíos. Se presiona el botón filtrar. El sistema muestra toda las noticias que están publicadas.	El sistema muestra todas las noticias que están publicadas.	Satisfactorio	Visor/ Clic en el botón Filtrar.
	EC 1.3: Filtrar noticias sin éxito.	El usuario introduce datos incorrectos. El sistema muestra un mensaje "No hay resultados".	El sistema muestra un mensaje "No hay resultados".	No Satisfactorio.	Visor/ Clic en el botón Filtrar.

Tabla 6 Caso de Prueba Filtrar noticias

Nombre de la sección	Escenarios de la sección	Descripción de la funcionalidad	Respuesta del sistema	Resultado de la prueba	Flujo Central
----------------------	--------------------------	---------------------------------	-----------------------	------------------------	---------------

CAPÍTULO III. VALIDACIÓN DE LA SOLUCIÓN PROPUESTA.

SC 1: Listar noticia	EC 1.1: Listar noticia con éxito.	El sistema una vez que carga la aplicación muestra un listado con todas las noticias que se encuentran publicadas en la base de datos en forma de imagen con el logo de la sección y el título.	El sistema muestra un listado con las noticias que están publicadas.	Satisfactorio.	Visor
	EC 1.2: Listar noticias sin éxito.	No existen noticias publicadas en la Base de datos. El sistema muestra un mensaje "No hay resultados".	El sistema muestra un mensaje "No hay resultados".	No Satisfactorio.	Visor

Tabla 7 Caso de Prueba Listar noticias

Nombre de la sección	Escenarios de la sección	Descripción de la funcionalidad	Respuesta del sistema	Resultado de la prueba	Flujo Central
----------------------	--------------------------	---------------------------------	-----------------------	------------------------	---------------

CAPÍTULO III. VALIDACIÓN DE LA SOLUCIÓN PROPUESTA.

SC 1: Visualizar noticia	EC 1.1: Visualizar éxito.	El sistema una vez que carga la aplicación muestra una interfaz con el nombre de la sección, título de la noticia y el tipo de pantalla de la misma (texto, texto-imagen, imagen y video). Además brinda las opciones de Pausa, Siguiente y Anterior.	El sistema muestra una interfaz con los datos de las noticias y las opciones de Siguiente, Pausa y Anterior.	Satisfactorio.	Visor
	EC 1.2: Visualizar éxito.	Una vez que cargue la aplicación no exista ninguna noticia publicada.	El sistema muestra la interfaz vacía.	No Satisfactorio.	Visor
	EC 1.3: Pausar	El usuario presiona el botón Pausar. El sistema detiene la reproducción de dicha noticia.	El sistema detiene la reproducción de dicha noticia.	Satisfactorio.	Visor/botón Pausar

CAPÍTULO III. VALIDACIÓN DE LA SOLUCIÓN PROPUESTA.

	EC 1.4: Anterior	El usuario presiona el botón anterior. El sistema muestra la noticia anterior.	El sistema muestra la noticia anterior.	Satisfactorio.	Visor /botón Anterior
	EC 1.5: Siguiete	El usuario presiona el botón siguiente. El sistema muestra la siguiente noticia o la siguiente pantalla.	El sistema muestra la siguiente noticia a visualizar o la próxima pantalla.	Satisfactorio.	Visor /botón Siguiete

Tabla 8 Caso de Prueba Visualizar noticias

Como resultado de la primera iteración de las pruebas se obtuvieron 6 no conformidades y en la segunda iteración las no conformidades fueron nulas. Por lo que se comprobó que las respuestas del sistema son las esperadas y que el mismo cumple con las especificidades planteadas y a su vez coinciden con las descripciones de los casos de uso planteados con anterioridad. Con las pruebas aplicadas se comprobó que el sistema cumple con los objetivos propuestos.

3.5 Conclusiones Parciales.

En este capítulo se realizó la validación de la solución propuesta. En el mismo se describieron los principales artefactos del modelo de implementación y despliegue, así como los resultados de las pruebas realizadas al visor web concluyendo que:

CAPÍTULO III. VALIDACIÓN DE LA SOLUCIÓN PROPUESTA.

- La elaboración de artefactos como el diagrama de despliegue y el modelo de implementación sirvió de guía para el desarrollo del módulo y permitió brindar una descripción más detallada de la estructura del sistema.
- Las pruebas de caja negra permitieron detectar, documentar y solucionar los errores existentes en el sistema implementado, obteniendo un software que cumple con todas las funcionalidades y calidad necesaria para su correcto funcionamiento.

REFERENCIAS BIBLIOGRÁFICAS

CONCLUSIONES GENERALES

Con el presente trabajo de diploma se logró dar cumplimiento a todas las tareas de la investigación propuestas inicialmente, obteniendo como resultado el visor web para la Plataforma de Televisión Informativa Primicia, llegando a las siguientes conclusiones:

- La caracterización de los procesos relacionados con la visualización de las noticias publicadas en el proyecto Primicia, permitió comprender cómo ocurre su flujo e implementar con mayor precisión las funcionalidades correspondientes.
- El análisis de las soluciones existentes permitió estudiar las características de cada una de ellas y extraer las que son comunes para el sistema a desarrollar.
- La caracterización, análisis y selección de las herramientas y tecnologías correctas, llevaron a elaborar una solución informática que fuera capaz de cumplir el objetivo trazado.
- El análisis del objeto de estudio y el campo de acción permitió darle solución al problema planteado de la carencia de un mecanismo que visualice las noticias publicadas en PRIMICIA.
- El trabajo de ingeniería realizado durante el desarrollo de la aplicación posibilitó documentar detalladamente los artefactos generados en los flujos de trabajo de la metodología seleccionada, con el objetivo de que los mismos sean utilizables en versiones futuras.
- Se ha obtenido como producto principal de este trabajo el visor web para el proyecto Primicia que cumple con las funcionalidades establecidas en el proyecto, entre las que se encuentran: visualizar noticia lo cual da solución a la problemática planteada de no existir un mecanismo que visualice las noticias publicadas y que permita la identificación de errores y de contenidos inadecuados.
- Con el desarrollo del visor web se logró dar solución al objetivo propuesto en la investigación.
- El uso de tecnologías y herramientas libres para el desarrollo del módulo permitió cumplir con la política de soberanía tecnológica por la que aboga el país.

REFERENCIAS BIBLIOGRÁFICAS

REFERENCIAS BIBLIOGRÁFICAS

1. definicion.de. [En línea] [Citado el: 15 de 10 de 2011.] [http://definicion.de/modulo/..](http://definicion.de/modulo/)
2. Diccionario de la Real Academia española. [En línea] [Citado el: 26 de 10 de 2011.] <http://buscon.rae.es/draeI/SrvltGUIBusUsual>.
3. LCC Lenguajes y Ciencias de la Computación. [En línea] [Citado el: 6 de 10 de 2011.] [http://www.lcc.uma.es/~pepeg/modula/temas/tema10.pdf..](http://www.lcc.uma.es/~pepeg/modula/temas/tema10.pdf)
4. Isi Lenguajes y Sistemas Informáticos. [En línea] [Citado el: 25 de 10 de 2011.] <http://www.isi.us.es/docencia/get.php?id=352>.
5. Hooping.net Negocios web. [En línea] [Citado el: 8 de 10 de 2011.] <http://www.hooping.net/glossary/aplicaciones-web-146.aspx>.
6. **alegsa**. alegsa. [En línea] [Citado el: 18 de 02 de 2012.] www.alegsa.com.ar/Dic/visor.php.
7. **Rodríguez, Felix Ivan Romero**. *Plataforma de televisión Informativa Primicia: Implementación del módulo de Redacción informativa PRIMICIA: Implementación del módulo de Redacción* . Ciudad de la Habana : Universidad de las Ciencias Informáticas. , 2011. s.n Tesis.
8. **BBC**. BBC Mundo. [En línea] [Citado el: 27 de 02 de 2012.] <http://www.bbc.co.uk/>.
9. **Sonaps**. Sonaps. [En línea] [Citado el: 29 de 01 de 2012.] <http://www.sony.fi/biz/lang/en/fi/product/network->.
10. **SONAPS**. SONAPS Sistema de Producción en Red. Sistema de Producción en Red. [En línea] [Citado el: 19 de 02 de 2012.] <http://www.scribd.com/doc/8328747/Sonaps-Sistema-de-Produccion-en-Red>.
11. **Enterprises, MECK**. eNewsRoom . eNewsRoom. [En línea] [Citado el: 03 de 02 de 2012.] <http://www.ereleases.com/pr/free-online-enevnewsroom>.
12. **Estructuretv**. Estructuretv. [En línea] [Citado el: 19 de 02 de 2012.] <http://www.estructuretv.com/enewsroom.htm>.
13. **Yuriana Araujo, Hilda Mendoza, Alexandre Torrealba, Luis German**, . scribd. [En línea] [Citado el: 15 de 03 de 2012.] <http://es.scribd.com/doc/31440864/Metodologia-RUP>.
14. *blogspot*. [En línea] [Citado el: 10 de 11 de 2011.] [http://metodologiaxpvsmetodologiarup.blogspot.com/2008/04/metodologia-rup.html..](http://metodologiaxpvsmetodologiarup.blogspot.com/2008/04/metodologia-rup.html)
15. **Juan Pablo Gómez Gallego, Ing. Jorge Galves**. **FUNDAMENTOS DE LA METODOLOGIA RUP**. . scribd. [En línea] [Citado el: 10 de 11 de 2011.] [http://es.scribd.com/doc/297224/RUP..](http://es.scribd.com/doc/297224/RUP)

REFERENCIAS BIBLIOGRÁFICAS

16. *maestrosdelweb*. [En línea] [Citado el: 8 de 11 de 2011.] [http://www.maestrosdelweb.com/editorial/el-framework-symfony-una-introduccion-practica-i-parte/...](http://www.maestrosdelweb.com/editorial/el-framework-symfony-una-introduccion-practica-i-parte/)
17. *symfony.es*. . [En línea] [Citado el: 4 de 11 de 2011.] <http://www.symfony.es/que-es-symfony/>.
18. **wordpress**. El rincón del un parásito virtual. [En línea] [Citado el: 5 de 1 de 2012.] <http://parasitovirtual.wordpress.com/2011/02/03/introduccion-a-symfony2/>.
19. *maestrosdelweb*. [En línea] [Citado el: 2011 de 11 de 2011.] <http://www.maestrosdelweb.com/editorial/%C2%BFque-es-javascript/>.
20. Jargonhosting Proveedor de soluciones en inetrnet. [En línea] [Citado el: 25 de 10 de 2011.] <http://www.jargon.com.ar/downloads/Lenguaje.HTML.pdf>.
21. *I.E.S Gaviota El language HTML*. [En línea] [Citado el: 17 de 10 de 2011.] <http://www.juntadeandalucia.es/averroes/iesgaviota/informatica/html.html>.
22. *maestrosdelweb*. [En línea] [Citado el: 6 de 11 de 2011.] <http://www.maestrosdelweb.com/editorial/phpintro/>.
23. *freedownloadmanager*. [En línea] [Citado el: 10 de 11 de 2011.] http://www.freedownloadmanager.org/es/downloads/Paradigma_Visual_para_UML_%5Bcuenta_de_Plataforma_de_Java_14715_p/..
24. *slideshare*. [En línea] [Citado el: 4 de 11 de 2011.] <http://www.slideshare.net/vanquishdarkenigma/visual-paradigm-for-uml..>
25. *magma*. [En línea] [Citado el: 10 de 11 de 2011.] http://www.magma.com.ni/~jorge/upoli_uml/refs/Que_es_UML.doc..
26. DocIRS. [En línea] [Citado el: 10 de 11 de 2011.] <http://www.docirs.cl/uml.htm..>
27. *babylon*. [En línea] [Citado el: 12 de 11 de 2011.] http://www.babylon.com/definicion/entorno_de_desarrollo/Spanish..
28. *showthread*. [En línea] [Citado el: 8 de 11 de 2011.] <http://foro.ignetwork.net/showthread.php?15188-IDE-Entorno-integrado-de-desarrollo-%28Concepto-importante%29...>
29. *Gadgetcabaret's weblog*. . [En línea] [Citado el: 12 de 11 de 2011.] <http://gadgetcabaret.wordpress.com/2007/12/05/netbeans-60-instalacion-y-caracteristicas>.
30. **Ciberaula España**. *Ciberaula*. [En línea] 2010. [Citado el: 2 de 12 de 2011.] http://linux.ciberaula.com/articulo/linux_apache_intro.
31. Apache. [En línea] The Apache Software Foundation., 2011. http://httpd.apache.org/docs/2.0/es/new_features_2_0.html.
32. *Postgresql-es*. [En línea] [Citado el: 1 de 12 de 2011.] http://www.postgresql.org.es/sobre_postg.
33. Ecured. [En línea] [Citado el: 1 de 12 de 2011.] <http://www.ecured.cu/index.php/PostGreSQL>.

REFERENCIAS BIBLIOGRÁFICAS

34. *desarrolloweb*. [En línea] [Citado el: 1 de 12 de 2011.] <http://www.desarrolloweb.com/manuales/manual-jquery.html> .
35. **Ecured**. Ecured. [En línea] [Citado el: 17 de 2 de 2012.] [http://www.ecured.cu/index.php/Modelo de dominio](http://www.ecured.cu/index.php/Modelo_de_dominio).
36. **Craig, Larman**. *UML Y PATRONES. Introducción al análisis y diseño orientado a objetos*. Mexico : Pearson, 1999.
37. Geek the planet. [En línea] [Citado el: 20 de 02 de 2012.] <http://geektheplanet.net/5462/patrones-gof.xhtml>..
38. **Ivar Jacobson, Grady Booch , Jame Rumbaugh**. *El Proceso unificado de desarrollo de software*. 2000.

BIBLIOGRAFÍAS

BIBLIOGRAFÍAS

1. definicion.de. [Online] [Cited: 10 15, 2011.] [http://definicion.de/modulo/..](http://definicion.de/modulo/)
2. Diccionario de la Real Academia española. [Online] [Cited: 10 26, 2011.] <http://buscon.rae.es/draeI/SrvltGUIBusUsual>.
3. LCC Lenguajes y Ciencias de la Computación. [Online] [Cited: 10 6, 2011.] [http://www.lcc.uma.es/~pepeg/modula/temas/tema10.pdf..](http://www.lcc.uma.es/~pepeg/modula/temas/tema10.pdf)
4. Isi Lenguajes y Sistemas Informáticos. [Online] [Cited: 10 25, 2011.] <http://www.isi.us.es/docencia/get.php?id=352>.
5. Hooping.net Negocios web. [Online] [Cited: 10 8, 2011.] <http://www.hooping.net/glossary/aplicaciones-web-146.aspx>.
6. **alegsa**. alegsa. [Online] [Cited: 02 18, 2012.] www.alegsa.com.ar/Dic/visor.php.
7. **Rodríguez, Felix Ivan Romero**. *Plataforma de televisión Informativa Primicia: Implementación del módulo de Redacción informativa PRIMICIA: Implementación del módulo de Redacción* . Ciudad de la Habana : Universidad de las Ciencias Informáticas. , 2011. s.n Tesis.
8. **BBC**. BBC Mundo. [Online] [Cited: 02 27, 2012.] <http://www.bbc.co.uk/>.
9. **Sonaps**. *Sonaps*. [Online] [Cited: 01 29, 2012.] <http://www.sony.fi/biz/lang/en/fi/product/network->.
10. **SONAPS**. SONAPS Sistema de Producción en Red. Sistema de Producción en Red. [Online] [Cited: 02 19, 2012.] <http://www.scribd.com/doc/8328747/Sonaps-Sistema-de-Produccion-en-Red>.
11. **Enterprises, MECK. eNewsRoom** . eNewsRoom. [Online] [Cited: 02 03, 2012.] <http://www.ereleases.com/pr/free-online-eneewsroom>.
12. **Estructuretv**. Estructuretv. [Online] [Cited: 02 19, 2012.] <http://www.estructuretv.com/enewsroom.htm>.
13. **Yuriana Araujo, Hilda Mendoza, Alexandre Torrealba, Luis German**, . scribd. [Online] [Cited: 03 15, 2012.] <http://es.scribd.com/doc/31440864/Metodologia-RUP>.
14. *blogspot*. [Online] [Cited: 11 10, 2011.] <http://metodologiexpvsmetodologiarup.blogspot.com/2008/04/metodologia-rup.html>.
15. **Juan Pablo Gómez Gallego, Ing. Jorge Galves. FUNDAMENTOS DE LA METODOLOGIA RUP**. . scribd. [Online] [Cited: 11 10, 2011.] <http://es.scribd.com/doc/297224/RUP>..
16. *maestrosdelweb*. [Online] [Cited: 11 8, 2011.] [http://www.maestrosdelweb.com/editorial/el-framework-symfony-una-introduccion-practica-i-parte/...](http://www.maestrosdelweb.com/editorial/el-framework-symfony-una-introduccion-practica-i-parte/)
17. *symfony.es*. . [Online] [Cited: 11 4, 2011.] <http://www.symfony.es/que-es-symfony/>.
18. **wordpress**. El rincon del un parásito virtual. [Online] [Cited: 1 5, 2012.] <http://parasitovirtual.wordpress.com/2011/02/03/introduccion-a-symfony2/>.

BIBLIOGRAFÍAS

19. maestrosdelweb. [Online] [Cited: 11 2011, 2011.] <http://www.maestrosdelweb.com/editorial/%C2%BFque-es-javascript/>.
20. Jargonhosting Proveedor de soluciones en inetrnet. [Online] [Cited: 10 25, 2011.] <http://www.jargon.com.ar/downloads/Lenguaje.HTML.pdf>.
21. I.E.S Gaviota El language HTML. [Online] [Cited: 10 17, 2011.] <http://www.juntadeandalucia.es/averroes/iesgaviota/informatica/html.html>.
22. maestrosdelweb. [Online] [Cited: 11 6, 2011.] <http://www.maestrosdelweb.com/editorial/phpintro/>.
23. freedownloadmanager. [Online] [Cited: 11 10, 2011.] http://www.freedownloadmanager.org/es/downloads/Paradigma_Visual_para_UML_%5Bcuenta_de_Plataforma_de_Java_14715_p/..
24. slideshare. [Online] [Cited: 11 4, 2011.] <http://www.slideshare.net/vanquishdarkenigma/visual-paradigm-for-uml..>
25. magma. [Online] [Cited: 11 10, 2011.] http://www.magma.com.ni/~jorge/upoli_uml/refs/Que_es_UML.doc..
26. DocIRS. [Online] [Cited: 11 10, 2011.] <http://www.docirs.cl/uml.htm..>
27. babylon. [Online] [Cited: 11 12, 2011.] http://www.babylon.com/definicion/entorno_de_desarrollo/Spanish..
28. showthread. [Online] [Cited: 11 8, 2011.] <http://foro.ignetwork.net/showthread.php?15188-IDE-Entorno-integrado-de-desarrollo-%28Concepto-importante%29...>
29. Gadgetcabaret 's weblog. [Online] [Cited: 11 12, 2011.] <http://gadgetcabaret.wordpress.com/2007/12/05/netbeans-60-instalacion-y-caracteristicas>.
30. Ciberaula España.Ciberaula. [Online] 2010. [Cited: 12 2, 2011.] http://linux.ciberaula.com/articulo/linux_apache_intro.
31. Apache. [Online] The Apache Software Foundation., 2011. http://httpd.apache.org/docs/2.0/es/new_features_2_0.html.
32. Postgresql-es. [Online] [Cited: 12 1, 2011.] http://www.postgresql.org.es/sobre_postg.
33. Ecured. [Online] [Cited: 12 1, 2011.] <http://www.ecured.cu/index.php/PostgreSQL>.
34. desarrolloweb. [Online] [Cited: 12 1, 2011.] <http://www.desarrolloweb.com/manuales/manual-jquery.html>
35. Ecured. Ecured. [Online] [Cited: 2 17, 2012.] <http://www.ecured.cu/index.php/Modelo de dominio>.
36. Craig, Larman.UML Y PATRONES. Introducción al análisis y diseño orientado a objetos. Mexico : Pearson, 1999.
37. Geek the planet. [Online] [Cited: 02 20, 2012.] <http://geektheplanet.net/5462/patrones-gof.xhtml..>

BIBLIOGRAFÍAS

38. **Ivar Jacobson, Grady Booch , Jame Rumbaugh.***El Proceso unificado de desarrollo de software.* 2000.
39. **Dustin Edward.** Scribd. [Online] [Cited: 04 20, 2012.] <http://es.scribd.com/doc/74894700/44/CAPITULO-4-IMPLEMENTACION-Y-PRUEBA-DEL-SISTEMA>.
40. **Guitierrez, Javier.** *¿Qué es un framework web?* . 2000.
41. **Hernández García, Ruber, et al.***Primicia, una Plataforma de Televisión sobre Tecnologías Libres.*
42. **Booch, Grady, Jacobson, Ivar and Rumbaugh, James.***El Proceso Unificado de Desarrollo de Software.* Madrid : Pearson Education, 2000.
43. **Pressman, Roger. S.***Ingeniería del Software. Un enfoque práctico.* 6ta. s.l. : McGraw-Hill, 2007.
44. **Sommerville, Ian.***Ingeniería del Software.* Madrid : Pearson Educación, 2002.

ANEXOS

ANEXOS

ANEXOS 1.

El diagrama siguiente describe detalladamente el framework JQuery, utilizado para lograr un mejor diseño de la interfaz y una rápida respuesta a los usuarios.

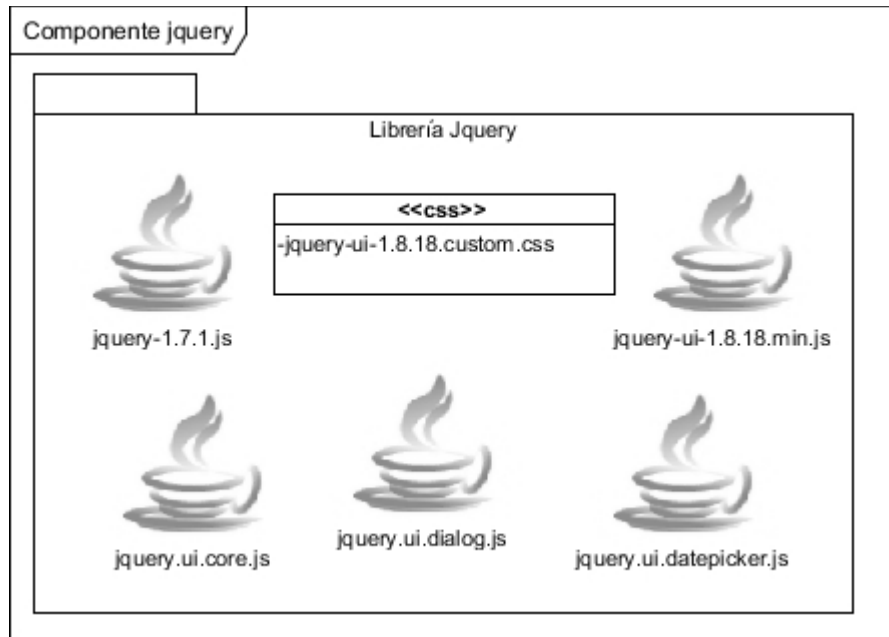


Figura 18 Componentes de JQuery

GLOSARIO DE TÉRMINOS

GLOSARIO DE TÉRMINOS.

Framework: Es una estructura de software integrada por componentes personalizables e intercambiables, en la cual otro proyecto de software puede ser organizado y desarrollado. Simplifican el desarrollo de las aplicaciones mediante la automatización de muchas de las tareas comunes. Además, un framework proporciona estructura al código fuente, forzando al programador a crear código más legible y más fácil de mantener. Ayudan en el desarrollo de software, proporcionan una estructura definida la cual ayuda a crear aplicaciones con mayor rapidez. Ayuda a la hora de realizar el mantenimiento del sitio gracias a la organización durante el desarrollo de la aplicación. Son desarrollados con el objetivo de brindarles a los programadores y diseñadores una mejor organización y estructura a sus proyectos.

Infocintas: Cintillo informativo que se muestra en la televisión, promocionando eventos de importancia, noticias relevantes que se transmitirán o información general para los televidentes.

Layout: Suele utilizarse para nombrar al esquema de distribución de los elementos dentro un diseño. Es habitual que un diseñador que se dedica a la creación de páginas web desarrolle un layout y se lo presente a su cliente para que éste lo apruebe y decida sobre la distribución de los contenidos. El layout, en este sentido, puede ser una especie de plantilla que presenta tablas o espacios en blanco. La idea es que, a partir del layout, la página web comience a desarrollarse con sus contenidos específicos.

Ortofotografía (del griego Orthós: correcto, exacto) es una presentación fotográfica de una zona de la superficie terrestre, en la que todos los elementos presentan la misma escala, libre de errores y deformaciones, con la misma validez de un plano cartográfico

Plantilla: A nivel general, una plantilla es un medio que posibilita portar o construir un diseño predefinido. Las plantillas facilitan la reproducción de copias idénticas, aunque también permiten crear algo nuevo cuando sólo actúan como un punto de partida o patrón. En el ámbito de la informática, una plantilla es un documento compuesto de líneas y tablas, con distintos márgenes y tamaños, que facilitan el desarrollo de páginas web, cartas u otros contenidos.

Plugins: Pequeño programa o módulo de hardware o software que añade una característica o un servicio específico a un sistema más grande, habitualmente de mayor tamaño. Un programa puede tener uno o más conectores. Son muy utilizados en los programas navegadores para ampliar sus funcionalidades.

Subsistema: Agrupación de componentes que hace parte de un sistema.

GLOSARIO DE TÉRMINOS

Teletexto. El teletexto es un servicio de información que muestra por la televisión las noticias en forma de texto combinado con imágenes, sonidos y videos.

TIC: Las Tecnologías de la Información y la Comunicación, también conocidas como TIC, son el conjunto de tecnologías desarrolladas para gestionar información y enviarla de un lugar a otro. Abarcan un abanico de soluciones muy amplio. Incluyen las tecnologías para almacenar información y recuperarla después, enviar y recibir información de un sitio a otro, o procesar información para poder calcular resultados y elaborar informes.

Twig: Motor de plantillas de symfony2 que permite separar el código PHP del código HTML en dos capas. La primera sería el programa con la lógica de negocio para resolver el problema específico de esa página, mientras que la otra sería una página que no contenga el mencionado código sino solo lo necesario para mostrar los datos a los usuarios. Presenta diferentes ventajas como herencia de plantilla y bloques, funciones específicas y extensible.

XML: Metalenguaje extensible de etiquetas desarrollado por el World Wide Web Consortium (W3C).