

Universidad de las Ciencias Informáticas

Facultad 4



**Universidad de las Ciencias
Informáticas**

**IMPLEMENTACIÓN Y PRUEBA DE LA VERSIÓN 1.1 DEL MÓDULO MEDIATECA DE LA COLECCIÓN
DE SOFTWARE EDUCATIVO “EL NAVEGANTE”**

**TRABAJO DE DIPLOMA PARA OPTAR POR EL TÍTULO DE INGENIERO EN CIENCIAS
INFORMÁTICAS**

AUTOR

EDYSNAY AYALA BALLESTER

TUTORES

ING. ANA MARÍA ÁLVAREZ VALDÉS

MsC. Prof. Asist. Lic. GRACIELA FALCÓN PI

CO-TUTOR

ING. RAFAEL JACOBO HIDALGO URBINO

LA HABANA, JUNIO 2012

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 ____ días del mes de junio del año 2012.

Autor:

Edysnay Ayala Ballester

Tutores:

Ing. Ana M. Álvarez Valdés

MsC. Prof. Asist. Lic. Graciela Falcón Pi

Cotutor:

Ing. Rafael Jacobo Hidalgo

RESUMEN

La Universidad de las Ciencias Informáticas (UCI) ha estado trabajando en base a la implementación de software educativos de alta calidad que constituyen hiperentornos educativos o de aprendizaje. Como parte del convenio integral de cooperación Cuba-Venezuela en la X mixta, se realizó un contrato para el desarrollo de una nueva versión de la colección “El Navegante”, existente en las escuelas secundarias cubanas, teniendo en cuenta las necesidades de Venezuela. Durante el curso 2009-2010 fue desarrollado el módulo Mediateca, perteneciente a dicha colección, cuya principal función es permitir la visualización de imágenes, así como la reproducción de sonidos, videos, animaciones y diaporamas de la colección, también facilitaría la consulta de palabras de difícil comprensión a través del glosario de términos y de datos de personalidades relacionadas con las temáticas abordadas en el software, sin embargo, no se cumplió con las expectativas del Ministerio del Poder Popular para la Educación venezolana producto a que se realizó antes del levantamiento de requisitos con el cliente. En estas condiciones surgió la necesidad de efectuar la implementación de la versión 1.1 del módulo Mediateca y validar la propuesta de solución mediante pruebas de Caja Negra que permitió erradicar deficiencias encontradas donde se presentó el diseño de los casos de pruebas y las no conformidades encontradas, las cuales fueron solucionadas. Se seleccionaron las tecnologías y herramientas necesarias para su desarrollo, así como, la elaboración del Diagrama de Despliegue y los Diagramas de Componentes que conforman el Modelo de Implementación. Donde se obteniendo un producto que cumplió con las necesidades del Ministerio del Poder Popular para la Educación de Venezuela y permitió una mayor interacción de los usuarios con los recursos multimedia de la colección.

PALABRAS CLAVE: módulo Mediateca, software educativo.

ÍNDICE

INTRODUCCIÓN	1
FUNDAMENTACIÓN TEÓRICA.....	6
1.1 DEFINICIONES RELACIONADAS AL PROCESO DE SOFTWARE EDUCATIVO.....	6
1.2 ANÁLISIS DE SISTEMAS SIMILARES EXISTENTES	8
1.2.1 COLECCIÓN MULTISABER	9
1.2.2 COLECCIÓN EL NAVEGANTE	9
1.2.3 COLECCIÓN FUTURO.....	9
1.3 TECNOLOGÍAS Y HERRAMIENTAS A UTILIZAR	10
1.3.1 PATRÓN ARQUITECTÓNICO MODELO-VISTA-CONTROLADOR.	10
1.3.3 LENGUAJES DE PROGRAMACIÓN.	11
1.3.4 FRAMEWORKS.	13
1.3.5 SERVIDOR WEB.	14
1.3.6 SISTEMA GESTOR DE BASE DE DATOS (SGBD).....	15
1.3.7 ENTORNO DE DESARROLLO INTEGRADO (IDE).....	16
1.4 CONCLUSIONES DEL CAPÍTULO	17
IMPLEMENTACIÓN DEL MÓDULO MEDIATECA.....	18
2.1 INTRODUCCIÓN.	18
2.2 DESCRIPCIÓN DEL MÓDULO MEDIATECA.....	18
2.3 MODELO DE IMPLEMENTACIÓN.....	23
2.3.1 DIAGRAMA DE DESPLIEGUE.	23
2.3.2 DIAGRAMA DE COMPONENTES.....	25
2.4 ESTÁNDAR DE CODIFICACIÓN.....	34

2.4.1 COMENTARIOS	34
2.4.2 IDENTACIÓN Y ESPACIOS EN BLANCO	35
2.4.3 CONVENCIONES DE NOMBRE (FUNCIONES, VARIABLES GLOBALES, CLASES).....	36
2.5 CONCLUSIONES DEL CAPÍTULO	38
PRUEBA DE SOFTWARE DEL MÓDULO MEDiateca.....	39
3.1 INTRODUCCIÓN	39
3.2 MÉTODOS DE PRUEBA.....	39
3.3 CLASIFICACIÓN DE LAS NO CONFORMIDADES	41
3.4 DISEÑO DE CASOS DE PRUEBA	42
3.5 RESULTADOS OBTENIDOS.....	66
3.6 CONCLUSIONES DEL CAPÍTULO	67
CONCLUSIONES GENERALES.....	68
RECOMENDACIONES	69
REFERENCIAS BIBLIOGRÁFICAS.....	70
BIBLIOGRAFÍA CONSULTADA.....	73
GLOSARIO DE TÉRMINOS.....	74

INTRODUCCIÓN

Durante los últimos años las Tecnologías de la Información y las Comunicaciones (TIC) han transformado a la sociedad, y en particular a los procesos educativos, los que han sufrido diversas modificaciones: cambios en la educación, en las relaciones interpersonales, en las formas de difundir y generar el conocimiento, cambios en los modos de intervención docente, desarrollo de nuevos espacios didácticos en línea, entre otros. Estos cambios generan grandes expectativas, controversias, demandas y desafíos en los sistemas de enseñanza y aprendizaje.

Internet ha introducido ya en gran parte de la población notables cambios en los hábitos y costumbres, debido a esto el debate sobre las TIC ya no puede mirarse solo como un recurso respecto al cual la escuela puede decidir si incluir o no, porque sus alumnos y docentes coexisten en este nuevo contexto cultural lo que supone un desarrollo de habilidades intelectuales de las cuales la escuela, no puede desentenderse. La incorporación de las nuevas tecnologías al ámbito educativo promueve la creación de nuevos entornos didácticos que afectan de manera directa tanto a los estudiantes como al escenario donde se lleva a cabo el proceso de enseñanza-aprendizaje.(1)

Con el objetivo de beneficiar la calidad de las clases en las escuelas cubanas, y así desarrollar el campo educativo, se han insertado con el transcurso de los años una serie de recursos informáticos, donde el software educativo es la principal herramienta informática de gran utilidad tanto para alumnos como para profesores.

Cuba, en particular la UCI, ha trabajado en base a la implementación de software educativos de alta calidad que constituyen hiperentornos educativos o de aprendizaje. El desarrollo de estos software le ha posibilitado la realización de contratos con empresas extranjeras y convenios de trabajo con empresas nacionales.

Como parte de estos contratos se han establecido relaciones con el Gobierno Bolivariano de Venezuela, que por su proceso revolucionario, se encuentra inmerso en constantes cambios en la esfera de la educación, por ello es sumamente importante la introducción de las TIC en esos centros educacionales.

En el convenio integral de cooperación Cuba-Venezuela en la X mixta, se realizó un contrato para el desarrollo del proyecto “Colecciones de Software Educativo Multisaber y El Navegante para los niveles

primaria y secundaria del sistema educativo venezolano“, con el objetivo de dotar al sistema escolar de la República Bolivariana de Venezuela de 24 software educativos, como herramientas pedagógicas en el proceso de enseñanza – aprendizaje.

La colección “El Navegante” cuenta con 10 productos que abarcan diferentes áreas del conocimiento que debe dominar un estudiante de secundaria básica; a su vez cada uno de ellos integra los módulos: General, Contenidos, Ejercicios, Juegos, Mediateca, Resultados y Maestro. En su conjunto todos estos módulos permiten organizar de forma modular la información que se le brinda al usuario y la implementación del software.

El módulo Mediateca tiene como objetivo permitir la visualización de imágenes, así como la reproducción de sonidos, videos, animaciones y diaporamas de la colección o de un tema¹ específico dentro de la misma. Además, Mediateca facilitaría la consulta de palabras de difícil comprensión a través del glosario de términos y de datos de personalidades relacionadas con las temáticas abordadas en el software.

Dicho módulo se desarrolló parcialmente antes de realizar el levantamiento de requisitos con el cliente, lo que, aunque significó un paso de avance, dio lugar a que no se cumpliera con las necesidades requeridas del Ministerio del Poder Popular para la Educación (en lo adelante MPPE) de Venezuela, donde quedó esta versión 1.0 con errores de funcionamiento y con las siguientes insuficiencias: no cuenta con un registro de las trazas de los estudiantes en las distintas secciones a las que accede; no brinda la posibilidad de consultar recursos multimedia por temas , no permite que se amplíen las imágenes seleccionadas en el módulo administrativo , no permite que se gestionen los recursos multimedia de forma efectiva y no superó la etapa de pruebas.

Este trabajo surge de la necesidad de dar solución a las problemáticas antes expuestas; por lo que se deriva el siguiente **problema científico**: ¿Cómo contribuir al perfeccionamiento del módulo Mediateca de la colección de software educativo “El Navegante” cumpliendo con las necesidades requeridas por el MPPE de Venezuela?

Del problema a resolver expresado anteriormente, se precisó como **objeto de estudio**: Proceso de desarrollo de software con tecnología multimedia.

¹ Contenido al cual van a estar relacionadas las medias.

Para ofrecer una solución favorable se ha trazado como **objetivo general** de este trabajo realizar la implementación y prueba de la versión 1.1 del módulo Mediateca de la colección de software educativo “El Navegante”.

El **campo de acción** queda enmarcado en el módulo Mediateca de la colección de software educativo “El Navegante”.

Para darle cumplimiento al objetivo general se definieron los siguientes **objetivos específicos**:

- ✓ Elaborar el marco teórico de la investigación.
- ✓ Implementar las funcionalidades de la versión 1.1 del módulo Mediateca de la colección “El Navegante”
- ✓ Realizar pruebas de software a la versión 1.1 del módulo Mediateca de la colección “El Navegante”.

Para guiar la investigación se plantea la siguiente **idea a defender**: Si se implementa y prueba la versión 1.1 del módulo Mediateca de la colección de software educativo “El Navegante” eliminando las insuficiencias de la versión 1.0, se obtendrá un producto que cumpla con las necesidades requeridas por el MPPE de Venezuela.

Para cumplir de forma satisfactoria con los objetivos descritos y resolver la situación problemática planteada, se definieron las siguientes tareas de investigación:

- ✓ Análisis de la versión 1.0 del módulo Mediateca y de los sistemas similares existentes.
- ✓ Identificación de las herramientas informáticas que permitan la implementación y prueba de la versión 1.1 del módulo Mediateca.
- ✓ Estructuración del modelo de implementación del módulo.
- ✓ Identificación, modelación e implementación de los componentes.
- ✓ Comprobación del funcionamiento del módulo dentro de la línea base del producto general.
- ✓ Definición de los casos de prueba.

- ✓ Aplicación de las pruebas de Caja Negra al módulo Mediateca.

Para obtener la información necesaria para el proceso investigativo se utilizaron los siguientes **métodos de investigación**:

Los **métodos teóricos** utilizados fueron: el método histórico-lógico y el método analítico-sintético, los cuales permitieron conocer el desarrollo y evolución de los sistemas similares que muestran recursos multimedia y extraer la información más relevante acerca de las metodologías, tecnologías y herramientas posibles a ser utilizadas en el desarrollo de este módulo.

El **método empírico** utilizado es la observación, con el objetivo de tomar las mejores prácticas de sistemas con la misma finalidad del que se va desarrollar, así como sus vulnerabilidades para convertirlas en ventajas de este nuevo producto.

Con el desarrollo del presente trabajo de diploma se logra:

- ✓ La integración del módulo Mediateca con el resto de los módulos de la colección de software educativo “El Navegante”.
- ✓ Mayor interacción de los usuarios con los recursos multimedia.
- ✓ Una mejor gestión de los recursos multimedia de la colección.
- ✓ Cumplir con todas las necesidades requeridas por el Ministerio del Poder Popular para la Educación de Venezuela.

ESTRUCTURA CAPITULAR.

Capítulo 1: Fundamentación teórica: representa la base teórica del presente trabajo, se refleja el estudio de los sistemas similares, las principales metodologías, herramientas y lenguajes que se seleccionan para dar solución a la problemática planteada.

Capítulo 2: Implementación de la versión 1.1 del módulo Mediateca: se describen en términos de componentes, las clases y subsistemas identificados anteriormente en el diseño realizado en el trabajo de diploma que antecede a esta tesis. Se crean los artefactos relacionados con la organización de los componentes y su distribución en los nodos de ejecución. Se reflejan las buenas prácticas de programación utilizadas y los estándares de codificación.

Capítulo 3: Realización de las pruebas de software a la versión 1.1 del módulo Mediateca: describe la estrategia trazada para desarrollar la validación del módulo mediante el método de prueba seleccionado. Refleja la realización de los casos de prueba, su aplicación al sistema y el resultado que se obtuvo con el empleo de los mismos.

FUNDAMENTACIÓN TEÓRICA

Este capítulo es el resultado de la búsqueda y el análisis de la información vinculada a los procesos de desarrollo de software educativo. Se abordan en él conceptos asociados como software educativo e hiperentorno de enseñanza–aprendizaje. También un análisis de sistemas similares existentes y se hace una descripción de las herramientas y tecnologías que son necesarias para una correcta implementación del software. Para este análisis se necesitó información obtenida, consultada y recopilada de una amplia bibliografía.

1.1 DEFINICIONES RELACIONADAS AL PROCESO DE SOFTWARE EDUCATIVO

La necesidad de introducir la tecnología en los procesos docentes se ha hecho evidente, donde desempeña un papel fundamental el uso del software educativo. A continuación se expone un breve repaso de las definiciones más comunes que se manejan en torno al desarrollo de este.

Software educativo

Se reconoce como software educativo a los programas didácticos que facilitan impartir el conocimiento a una audiencia, además de adaptarlo al ritmo de trabajo y las necesidades de cada educando, para lograr diversas alternativas de aprendizaje. (2)

Según Rodríguez Lamas, software educativo es una aplicación informática, que soportada sobre una bien definida estrategia pedagógica, apoya directamente el proceso de enseñanza–aprendizaje constituyendo un efectivo instrumento para el desarrollo educacional del hombre del próximo siglo. (3)

Los software educativos pueden tratar diferentes materias de formas muy diversas, y ofrecer un entorno de trabajo más o menos sensible a las circunstancias de los alumnos y más o menos rico en posibilidades de interacción. Están caracterizados por ser altamente interactivos a partir del empleo de recursos multimedia. (4)

Recursos multimedia

Los componentes multimedia son los diversos medios de comunicación digitalizados que tiene el hombre para transmitir un mensaje con imágenes de cualquier tipo, sonidos, películas de video, y hasta el mismo

texto, y constituyen un elemento crucial en la calidad de los software educativos, lo que posibilita que esta herramienta de apoyo al proceso de enseñanza - aprendizaje sea más atractiva e interactiva.

La tecnología multimedia permite el acceso, a través de la computadora, a información de diversa naturaleza (una película, un sonido, una imagen, un texto), que permite la posibilidad de interactuar de manera dinámica con dichos recursos, donde supera la rigidez del sistema lineal que solo permitía transitar entre distintos menús. De este modo, habilita a aproximarse cada vez más a los modos del pensamiento humano, en virtud de la posibilidad de asociar, elegir y acceder a la información almacenada en la computadora. (5)

La interactividad es, en este caso, la que potencia su uso para la adquisición de saberes y el aprendizaje de estrategias que habilitan al estudiante en el conocimiento del sistema y consolidan sus propias habilidades.

Hipermedia

La tecnología hipermedia es ampliamente utilizada para la producción de aplicaciones informáticas con disímiles propósitos. Su empleo con fines docentes gana cada vez más espacio en los currículos de los diferentes niveles de enseñanza a nivel internacional y es ya una realidad su presencia en las escuelas cubanas.

Un sistema hipermedia es una estructura similar a la del hipertexto (organiza la información en forma de red, entendiendo esta como un modo de unión entre los elementos de información y se caracteriza porque no existe un orden único de recorrido entre ellos, no tiene principio ni fin), con la particularidad de que esta información es multimedia (textos, imágenes, secuencias de animaciones, sonidos, video). En esta red de información se distinguen tres elementos fundamentales: los nodos, que son los contenedores de la información; las anclas que indican las zonas de información que pueden estar relacionadas con otras; y los enlaces, que representan relaciones entre la información que contienen los nodos. (6)

En una hipermedia se pueden utilizar dos formas básicas para consultar la información que se encuentra almacenada en sus nodos: la navegación y la búsqueda. La navegación es considerada la forma primaria de acceso a la información en una hipermedia y consiste en recorrer los nodos a través de los enlaces, consultando la información que en él se almacena. En la búsqueda el usuario solicita al sistema el acceso directo a un determinado dato. (6)

Hiperentorno de enseñanza–aprendizaje

La colección “El Navegante” puede ser clasificada como un hiperentorno de enseñanza–aprendizaje, siendo este uno de los tipos de software educativo más difundido en el ámbito escolar contemporáneo, y se basa en la tecnología hipermedia.

Los hiperentornos de enseñanza–aprendizaje son los sistemas hipermedia que están concebidos para ser utilizados como medios de enseñanza–aprendizaje por profesores y estudiantes. Estos hiperentornos, que tienen en común la utilización de un ambiente informático con una intención didáctica, integran los agentes humanos (estudiantes, profesores) y los artificiales (recursos informáticos) y ofrecen condiciones de interacción local o a través de redes.

El concepto de hiperentorno de enseñanza–aprendizaje es, como plantea Pierre Tchounikine, lo suficientemente general para incluir tanto una plataforma de educación abierta o a distancia destinada a la difusión de recursos pedagógicos como un software para el aprendizaje de una determinada competencia de álgebra. (7) Los hiperentornos de enseñanza–aprendizaje se distinguen por las amplias potencialidades tecnológicas que poseen: (6)

- ✓ La integración de varias formas de presentación de la información (textos, imágenes, sonidos, animaciones, videos).
- ✓ Las posibilidades de interactividad de los estudiantes con los contenidos que le permiten el control de su aprendizaje.
- ✓ Las herramientas de comunicación que se disponen para el trabajo colaborativo.
- ✓ Las capacidades de almacenamiento de información sobre el trabajo realizado que permiten evaluar los resultados con carácter procesal.

1.2 ANÁLISIS DE SISTEMAS SIMILARES EXISTENTES

A medida que avanza la informatización de la sociedad cubana se introducen en las escuelas recursos informáticos con el propósito de beneficiar la calidad de las clases en las aulas. En este apartado se abordará sobre tres colecciones con que cuenta la escuela cubana: “Multisaber”, “El Navegante” y “Colección Futuro”, donde se tratan específicamente estas tres pues tienen integrado un módulo dedicado a la interacción del usuario con recursos multimedia.

1.2.1 COLECCIÓN MULTISABER

La colección Multisaber es un grupo de multimedia educativas destinadas a la enseñanza primaria. Está compuesta por 32 software e incluye en su modelo pedagógico seis módulos: Temas, Ejercicios, Biblioteca, Profesor, Resultados y Juegos. Multisaber posee una presentación genérica y una particular de cada producto. El nombre Multisaber para la colección expresa la idea del aprendizaje de múltiples contenidos para los niños desde temprana edad. Los software de Multisaber poseen una interfaz estandarizada, que proporcionan un ambiente de trabajo amigable e intuitivo con alto nivel de interactividad para hacer más amena la enseñanza y más efectivo el aprendizaje en los estudiantes. El módulo Biblioteca es el encargado de visualizar recursos multimedia (imágenes, videos, glosario), no permitiendo adicionar, eliminar y modificar ninguno de ellos.

1.2.2 COLECCIÓN EL NAVEGANTE

La versión de la colección “El Navegante” instalada en las escuelas secundarias cubanas está conformada por 10 productos y a su vez cada uno de ellos está compuesto por seis módulos básicos. Entre ellos existe un módulo “Mediateca”, encargado de la interacción del usuario con los recursos multimedia de la colección. La colección se caracteriza por el empleo de estrategias pedagógicas de análisis de respuestas y retroalimentaciones reflexivas. Registra la actuación de los estudiantes con el software para su empleo en el análisis de los resultados obtenidos usados para la toma de decisiones en la dirección del proceso pedagógico.

1.2.3 COLECCIÓN FUTURO

La colección Futuro está destinada a la educación preuniversitaria y contiene un total de 15 software que abarcan todas las asignaturas que se estudian en ese nivel. Los software que integran la colección no solo abarcan las asignaturas que se imparten a los estudiantes en las diferentes áreas de conocimiento, sino también materiales dirigidos a la preparación de profesores y directivos. Su concepción metodológica se ha denominado hiperentorno de aprendizaje y está formada por 6 módulos: Temas, Ejercicios, Biblioteca, Profesor, Juegos y Resultados. El módulo Biblioteca es un repositorio de recursos multimedia en el que aparecen: galerías de diversa naturaleza (imágenes, videos, sonidos, diaporamas, etc.), elementos de conectividad y efemérides.

Algunos de los inconvenientes que presentan dichos software son: no están implementados sobre estándares educativos, el funcionamiento de cada uno de sus productos no es el mismo, están

desarrollados con herramientas propietarias, no son multiplataforma lo que limita su utilización en otros sistemas operativos y no permiten que se gestionen recursos multimedia.

1.3 TECNOLOGÍAS Y HERRAMIENTAS A UTILIZAR

Teniendo en cuenta que las tecnologías y herramientas utilizadas en el desarrollo de los productos que conforman la colección “El Navegante” se encuentran definidas en el documento de arquitectura del proyecto “Arquitectura de la versión multiplataforma de la colección de software educativo El Navegante”, se analiza a continuación las características fundamentales que propiciaron su elección y los beneficios que aportan al módulo Mediateca, sustentado también en la investigación realizada para el trabajo de diploma “Análisis y Diseño de la versión 1.1 del módulo Mediateca de la colección El Navegante” en el que se definió la metodología de desarrollo, el lenguaje y la herramienta de modelado a utilizar para la realización del presente trabajo, tomando como metodología de desarrollo de software, *Rational Unified Process* (RUP) el lenguaje de modelado *Unified Modeling Language* (UML) y como herramienta para el modelado Visual Paradigm.

1.3.1 PATRÓN ARQUITECTÓNICO MODELO-VISTA-CONTROLADOR.

Según su definición oficial en la IEEE5 Std 1471-2000 “La Arquitectura del Software es la organización fundamental de un sistema formada por sus componentes, las relaciones entre ellos y el contexto en el que se implantarán, y los principios que orientan su diseño y evolución”. (10)

La arquitectura no está atada a ninguna metodología de desarrollo, sino que es en sí una disciplina que evoluciona durante todo el ciclo de desarrollo de software. Abarca todo el proceso de desarrollo para dar distribución y orden a los elementos de un sistema informático.

Los patrones de arquitectura son modelos que proporcionan reglas y guías para organizar un sistema de software, permitiéndole a los programadores, diseñadores, ingenieros y analistas trabajar bajo una línea común que les posibilite la compatibilidad necesaria para lograr el objetivo deseado.

Teniendo en cuenta que la aplicación que se desea crear es interactiva con una interfaz humano-computadora flexible, se determinó que la mejor manera de estructurar el sistema es utilizando el patrón Modelo Vista Controlador o Model View Controller (MVC).

MVC (por sus siglas en inglés) es un patrón de diseño de arquitectura de software usado principalmente en aplicaciones complejas que manejan gran cantidad de datos, fue diseñado para reducir el esfuerzo de programación necesario en la implementación de sistemas.

Este modelo de arquitectura presenta varias ventajas: (11)

- ✓ Hay una clara separación entre los componentes de un programa; permitiendo implementarlos por separado.
- ✓ Facilita agregar nuevos tipos de datos según sea requerido por la aplicación ya que son independientes del funcionamiento de las otras capas.
- ✓ Facilita el mantenimiento en caso de errores.
- ✓ Facilita la funcionalidad, mantenibilidad y escalabilidad del sistema.
- ✓ La conexión entre el modelo y sus vistas es dinámica; se produce en tiempo de ejecución, no en tiempo de compilación.

Al incorporar el modelo de arquitectura MVC a un diseño, las piezas de un programa se pueden construir por separado y luego unirlas en tiempo de ejecución. Si uno de los componentes, posteriormente, se observa que funciona mal, puede reemplazarse sin que las otras piezas se vean afectadas. (11)

Symfony implementa este patrón arquitectónico con sus peculiaridades, el controlador está compuesto como mínimo, por un controlador frontal que controla las peticiones y se las envía a las clases Actions para que las procesen, el modelo cuenta con las clases del modelo o acceso a datos y una capa de abstracción de base de datos en este caso el ORM Doctrine. Por último, la vista tiene las plantillas necesarias para mostrar la información al usuario y como mínimo una plantilla global (layout) que encapsula el código HTML común de las demás plantillas.

1.3.3 LENGUAJES DE PROGRAMACIÓN.

Se hace necesario el uso de varios lenguajes de programación pues permiten crear programas mediante un conjunto de comandos, operadores y reglas de sintaxis que ponen a disposición del programador para que este pueda comunicarse con los dispositivos hardware y software existentes.

Los lenguajes del lado del cliente son aquellos que pueden ser directamente "digeridos" por el navegador y no necesitan un pre-tratamiento. Para el desarrollo de la aplicación se usarán:

XHTML

eXtensible Hypertext Markup Language (Lenguaje de Marcado de Hipertexto Extensible), extiende HTML 4.0 combinando la sintaxis de HTML (diseñado para mostrar datos) con la de XML (diseñado para describir los datos).

XHTML, está orientado al uso de un etiquetado correcto, exige una serie de requisitos básicos a cumplir en lo que a código se refiere. Entre estos requisitos básicos se puede mencionar, una estructuración coherente dentro del documento donde se incluirían elementos correctamente anidados, etiquetas en minúsculas, elementos cerrados correctamente, atributos de valores entrecomillados.

JavaScript 1.4

Es un lenguaje multiplataforma interpretado por la mayoría de los navegadores en sus últimas versiones por lo que no es necesario compilar los programas para ejecutarlos, es decir, los programas escritos con este lenguaje se pueden probar directamente en cualquier navegador sin necesidad de procesos intermedios. Es pequeño y ligero, no es útil como un lenguaje independiente, más bien está diseñado para una fácil incrustación en otros productos y aplicaciones, tales como, los navegadores web que permiten el desarrollo de interfaces de usuario mejoradas y páginas web dinámicas. Se trata de un lenguaje de tipo script compacto, basado en objetos (no orientado a objeto por lo que no dispone de herencia) y guiado por eventos. Los programas JavaScript van incrustados en los documentos HTML, y se encargan de realizar acciones en el cliente, como pueden ser: pedir datos, confirmaciones, mostrar mensajes, crear animaciones, comprobar campos. Es muy utilizado por su fiabilidad y seguridad ya que los scripts tienen capacidades limitadas. Es ejecutado en el cliente por lo que el servidor no es utilizado más de lo debido y como desventaja el código del script debe descargarse completamente para poder ejecutarlo.

CSS 3

Cascading Style Sheets Hojas de Estilo en Cascada: Es un lenguaje de hojas de estilos creado para controlar el aspecto o presentación de los documentos electrónicos definidos con HTML y XHTML. CSS es la mejor forma de separar la presentación de sus contenidos y es imprescindible para crear páginas web complejas. CSS permite definir estilos propios en un archivo externo a las páginas; así, si en algún

momento se quiere cambiar alguno de ellos, automáticamente se actualizarán todas las páginas vinculadas al sitio.

Los lenguajes del lado del servidor son aquellos lenguajes reconocidos, ejecutados e interpretados por el propio servidor y que se envían al cliente en un formato comprensible. Para esta investigación se decidió utilizar:

PHP 5.3

Hypertext Preprocessor: Lenguaje muy utilizado para la generación de sitios web dinámicos, ejecutados en el servidor. No necesita ser compilado para ejecutarse. Soporta en cierta medida la orientación a objeto. Se caracteriza por ser un lenguaje muy rápido, multiplataforma, capaz de conectarse con la mayoría de los manejadores de base de datos y es libre, por lo que se presenta como una alternativa de fácil acceso para todos. Como desventaja se puede mencionar que para la realización de proyectos complejos requiere un alto nivel de experiencia, y que por determinadas limitaciones del lenguaje existe un pequeño grupo de tareas que resultan muy difíciles de realizar en el mismo. (12)

1.3.4 FRAMEWORKS.

Como su nombre lo indica, entregan una estructura, almacén o marco para desarrollar proyectos. Persiguen acelerar el proceso de desarrollo, reutilizar código ya existente y promover buenas prácticas de implementación. Permite a los programadores pasar más tiempo identificando requerimientos que tratando los tediosos detalles de bajo nivel. La mayoría de los frameworks implementan uno o más patrones de diseño de software para asegurar la escalabilidad del producto. (13)

Es necesario entender que un framework es una abstracción de código común que provee funcionalidades genéricas que pueden ser utilizadas para desarrollar aplicaciones de manera rápida, fácil, modular y sencilla, ahorrando tiempo y esfuerzo. (14)

Symfony 1.4.3

Proporciona varias herramientas y clases encaminadas a reducir el tiempo de desarrollo de una aplicación web compleja. Separa la lógica de negocio, la lógica de servidor y la presentación de la aplicación web. Además, automatiza las tareas más comunes, que permite al desarrollador dedicarse por completo a los aspectos específicos de cada aplicación. A su favor tiene que es fácil de instalar y configurar en sistemas Linux, funciona con todas las bases de datos comunes, se adapta con facilidad a las políticas y

arquitecturas propias de cada empresa u organización. Se hará uso de su versión 1.4 la que automatiza la mayoría de elementos comunes de los proyectos web.

Symfony es el framework más documentado del mundo, cuenta con miles de páginas de documentación distribuidas en varios libros gratuitos y decenas de tutoriales. Aumenta exponencialmente la productividad y ayuda a mejorar la calidad de las aplicaciones web, aplicando todas las buenas prácticas y patrones de diseño que se han definido para la web. (15)

jQuery 1.5

Es una biblioteca JavaScript muy rápida y muy ligera que simplifica la manera de interactuar con los documentos HTML y el desarrollo del lado del cliente de las aplicaciones web. Las aplicaciones realizadas con jQuery pueden responder de forma mucho más rápida que las aplicaciones JavaScript tradicionales. (16)

Esta potente biblioteca encabeza las listas de bibliotecas JavaScript más utilizadas gracias a su sencillez, su flexibilidad, su potencia y su rendimiento. Es un producto serio, estable, bien documentado, con un gran equipo de desarrolladores a cargo de su mejora y actualización, muy aceptado por los programadores y con un grado de penetración en el mercado muy amplio, lo que invita a su uso como una de las mejores opciones para el desarrollo. (17)

1.3.5 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, donde entrega como resultado una página web o información de todo tipo de acuerdo con los comandos solicitados.

Apache 2.0

Es un servidor web Open Source (Código Abierto) muy utilizado a nivel mundial, es multiplataforma, extensible y posee una amplia documentación. Es un software que está estructurado en módulos, está dividido en muchas porciones de código que hacen referencia a diferentes aspectos o funcionalidades del servidor web.

Características generales:

- ✓ Es robusto y seguro.
- ✓ Posee una configuración simple y potente a través de ficheros.
- ✓ Tiene soporte para autenticación HTTP.
- ✓ Cuenta con un sistema de notificación de errores HTTP altamente configurable.
- ✓ Posibilita el chequeo avanzado de sintaxis de la *URL*, incluyendo caracteres falsos que son fuentes frecuentes de ataques como SQL Injection, CommandInjection y XSS.

1.3.6 SISTEMA GESTOR DE BASE DE DATOS (SGBD).

Un SGBD es una colección de programas cuyo objetivo es servir de interfaz entre la base de datos, el usuario y las aplicaciones. Se compone de un lenguaje de definición de datos, de uno de manipulación de datos y de otro de consulta. Un SGBD permite definir los datos a distintos niveles de abstracción y manipular dichos datos, donde garantiza la seguridad e integridad de los mismos.

PostgreSQL 8.4

Es un SGDB relacional libre orientado a objetos, está diseñado para administrar grandes cantidades de datos. Se ejecuta en los Sistemas Operativos más utilizados en el mundo, incluyendo Linux, varias versiones de UNIX y Windows. Incluye características de la orientación a objetos, como puede ser la herencia, tipos de datos, funciones, restricciones, disparadores, reglas e integridad transaccional.

Características fundamentales:

- ✓ Soporta distintos tipos de datos. Además de los más básicos, también soporta datos de tipo fecha, monetarios, elementos gráficos, arreglos, y datos sobre redes.
- ✓ Incorpora funciones de diversa índole: manejo de fechas, geométricas, orientadas a operaciones con redes.
- ✓ Permite la declaración de funciones propias, así como la definición de disparadores.
- ✓ Soporta el uso de índices, reglas y vistas.

- ✓ 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.
- ✓ Permite la gestión de diferentes usuarios, como también los permisos asignados a cada uno de ellos.

1.3.7 ENTORNO DE DESARROLLO INTEGRADO (IDE).

IDE acrónimo de (Integrated Development Environment) es un software compuesto por un conjunto de herramientas de programación que puede dedicarse en exclusiva a un sólo lenguaje de programación o varios. Los IDEs pueden ser aplicaciones por si solos o pueden ser parte de aplicaciones existentes. Están compuestos por un editor de texto, un compilador, un intérprete, herramientas de automatización y un depurador.

Netbeans IDE 6.9

Netbeans es un IDE de código abierto, gratuito sin restricciones de uso y una plataforma de aplicaciones que permiten a los desarrolladores crear de manera rápida diferentes tipos de aplicaciones. Es una herramienta realizada en java pensada para escribir, compilar, depurar y ejecutar programas de diferentes lenguajes, ya que cuenta con un número importante de módulos.

En el caso del NetBeans 6.9 brinda soporte para PHP 5.3 así como también para Symfony 1.4.3.

Algunas de las posibilidades que hasta ahora brinda el IDE son:

- ✓ Soporte para PHP 5.3
- ✓ Soporte para las palabras clave (*keywords*).
- ✓ Soporte para *namespaces*.
- ✓ Soporte para *closures*.
- ✓ Soporte para Symfony
- ✓ Ejecución de comando Symfony.
- ✓ Limpiar la caché.
- ✓ Ayuda para encontrar comandos y para conocer los parámetros.

- ✓ Muestra toda la ejecución de cada comando (creación de ficheros, directorios, etc.)

1.4 CONCLUSIONES DEL CAPÍTULO

En este capítulo se analizaron los sistemas similares que tienen un módulo encargado de visualizar recursos multimedia, se expusieron las principales características de las herramientas y tecnologías seleccionadas. Se ha decidido utilizar como lenguaje de programación PHP 5.3 por la amplia documentación que existe en la universidad y en Internet, la flexibilidad y la claridad de su código , además se ajusta a los requisitos de la aplicación, apache como servidor web y PostgreSQL8.4 como gestor de bases de datos formando una robusta plataforma de trabajo . Symfony1.4.3 como *framework* por la gran cantidad de código que genera automáticamente, por la facilidad de aprendizaje, el trabajo con los formularios y las validaciones y como entorno de desarrollo integrado NetBeans para el trabajo con HTML, JavaScript1.4, CSS 3 y PHP 5.3. Este IDE provee módulos y componentes para el trabajo con Symfony1.4.3 y facilita el trabajo en equipo, además es completamente libre y tiene versiones para varios sistemas operativos.

IMPLEMENTACIÓN DEL MÓDULO MEDIATECA

2.1 INTRODUCCIÓN.

La implementación comienza con el resultado del diseño y se implementa en términos de componentes, es decir ejecutables, ficheros de código fuente, librerías y archivos. Este capítulo describe cómo los elementos del diseño se implementan en términos de componentes y cómo estos se organizan respecto a los nodos específicos en el diagrama de despliegue. Se presentan también los requisitos del sistema y la adopción de un estándar de codificación.

2.2 DESCRIPCIÓN DEL MÓDULO MEDIATECA.

Este módulo tiene la finalidad de gestionar recursos multimedia como son imágenes, videos, sonidos, animaciones, diaporamas, glosario, personalidades y permite la visualización de imágenes, así como la reproducción de sonidos, videos, animaciones y diaporamas de la colección o de un tema específico dentro de la misma. Además, facilitaría la consulta de palabras de difícil comprensión a través del Glosario de términos y de datos de personalidades relacionadas con las temáticas abordadas en el software, para cumplir con el objetivo de este módulo fueron identificados, los siguientes requisitos del sistema.

Requisitos Funcionales (RF).

RF1. Mostrar la pantalla principal del módulo Mediateca.

RF1.1: Mostrar las imágenes que representan metafóricamente a cada una de las secciones que integran la Mediateca.

RF2. Mostrar cada una de las secciones que integran la Mediateca: videos, imágenes, sonidos, glosario, diaporamas, personalidades y animaciones.

RF2.1: Mostrar los elementos de videos, imágenes, sonidos, diaporamas y animaciones agrupados por temas.

RF3. Consultar un tema específico.

RF4. Reproducir videos, sonidos, animaciones y diaporamas.

RF4.1: Poner pausa.

RF4.2: Reproducir.

RF4.3: Rebobinar.

RF4.4: Adelantar.

RF4.5: Regular el volumen: aumentar, disminuir o silenciar.

RF5. Ver imagen seleccionada.

RF6. Ampliar a pantalla completa la media visualizada.

RF7. Ver la fuente de donde proviene la media visualizada.

RF8. Permitir la navegabilidad entre las pantallas internas de sección a través de las opciones de “atrás”, “siguiente”, “primera” y “última”.

RF9. Permitir en las secciones glosario y personalidades, buscar un elemento a través de la selección de una de las letras mostradas en la pantalla, o introduciendo por el teclado el nombre del elemento deseado.

RF9.1: Mostrar el contenido (texto, imagen) del elemento seleccionado por el usuario en una nueva ventana.

RF10. Mostrar el número de la página donde se encuentra el usuario y el total de páginas existentes en la sección o tema visible.

RF11. Mostrar el total de elementos y el elemento visualizado.

RF12. Permitir ir a la interfaz principal de la Mediateca desde cualquier pantalla donde se encuentre el usuario.

RF13. Gestionar videos.

RF13.1: Adicionar video.

RF13.2: Modificar datos del video.

RF13.3: Eliminar video.

RF14. Gestionar imágenes.

RF14.1: Adicionar imagen.

RF14.2: Modificar datos de la imagen.

RF14.3: Eliminar imagen.

RF15. Gestionar sonidos

RF15.1: Adicionar sonido.

RF15.2: Modificar datos del sonido.

RF15.3: Eliminar sonido.

RF16. Gestionar personalidades.

RF16.1: Adicionar personalidad.

RF16.2: Modificar datos de la personalidad.

RF16.3: Eliminar personalidad.

RF17. Gestionar diaporamas.

RF17.1: Adicionar diaporama.

RF17.2: Modificar datos del diaporama.

RF17.3: Eliminar diaporama.

RF18. Gestionar animaciones.

RF18.1: Adicionar animación.

RF18.2: Modificar datos de la animación.

RF18.3: Eliminar datos de la animación.

RF19. Gestionar glosario.

RF19.1: Adicionar término al glosario.

RF19.2: Modificar término del glosario.

RF19.3: Eliminar término del glosario.

RF20. Registrar la traza de los estudiantes en la Galería de imágenes.

RF21. Registrar la traza de los estudiantes en la Galería de videos.

RF22. Registrar la traza de los estudiantes en el Glosario de términos.

Listado de los Requisitos no Funcionales (RNF)

RNF1. Requisitos de software.

1. Computadora Personal con navegador Mozilla Firefox 3.5 o superior.

RNF2. Requisitos de hardware.

- ✓ Procesador Pentium 500 MHz o superior.
- ✓ Al menos 256 MB de RAM.
- ✓ Al menos 10 GB de espacio en disco duro.
- ✓ Dispositivos de audio.
- ✓ Soporte de video que admita resolución de al menos 800x600 y 24 bits.
- ✓ Dispositivo de red de al menos 10 Mbits.
- ✓ El sistema interactuará con una impresora que permita imprimir los diferentes contenidos como respuesta a las funcionalidades del sistema.

RNF3. Requisitos de restricciones en el diseño y en la implementación.

- ✓ Framework del lado del cliente: jQuery 1.4.

- ✓ Framework del lado del servidor: Symphony 1.4.3.
- ✓ Lenguaje de programación del lado del cliente: JavaScript.
- ✓ Lenguaje de programación del lado del servidor: PHP 5.2.
- ✓ Servidor web: Apache 2.0.
- ✓ IDE de desarrollo: NetBeans 6.9.
- ✓ Servidor de base de datos: PostgreSQL 8.4.
- ✓ Patrón arquitectónico: Modelo Vista Controlador.

RNF4. Requisitos de apariencia e interfaz externa.

- ✓ Las interfaces se desarrollarán teniendo en cuenta elementos de las aplicaciones con tecnología multimedia diseñadas para escritorio y las aplicaciones web.
- ✓ Las interfaces de usuario estarán acorde a las edades a las que van dirigidos cada uno de los 10 productos que integran la colección “El Navegante”.
- ✓ El sistema proporcionará claridad y correcta organización de la información, donde permite la interpretación correcta e inequívoca de esta.
- ✓ El diseño de la interfaz gráfica deberá garantizar la distinción visual entre los elementos del sistema.

RNF5. Requisitos de soporte.

- ✓ Se realizará transferencia tecnológica de la colección a los clientes.
- ✓ Se impartirán clases a los profesores venezolanos para explicar el funcionamiento y utilidad del producto.

RNF6. Requisitos de portabilidad.

- ✓ El sistema podrá ser utilizado bajo cualquier sistema operativo que soporte Mozilla Firefox.

RNF7. Requisitos de usabilidad.

- ✓ Facilidad de desarrollo con software libre debido a la amplia gama de herramientas disponibles para el desarrollo de aplicaciones web

- ✓ La utilización del software por parte de los estudiantes u otros usuarios sin avanzados conocimientos de computación, no requiere de previa preparación, debido al diseño sencillo y estándar que tiene.
- ✓ Cada producto cuenta con una ayuda que se ubicará como un servicio en el mismo, cubriendo cada una de las pantallas o conceptos del software y será escrita en lenguaje español lo más sencilla y asequible posible para facilitar su entendimiento por parte de los usuarios.

RNF8. Requisitos Legales.

- ✓ Al desplegarse el producto, todas las medias que hayan sido utilizadas deben tener el permiso legal de sus autores y su aprobación para su uso. Además deben respetarse las licencias y condiciones legales bajo las que fueron adquiridas.

RNF9. Requisitos de rendimiento.

Uso local

- ✓ Tiempo de respuesta de una página: no mayor a 30 segundos. Uso en la red
- ✓ Velocidad de respuesta máxima: 5 minutos.

2.3 MODELO DE IMPLEMENTACIÓN.

En la implementación se describen las clases y subsistemas del diseño en términos de componentes y como resultado de esta actividad se obtiene el Diagrama de Componentes, que refleja la organización física de estos y la relación que presentan entre sí, también cómo estos se organizan de acuerdo a los nodos específicos en el modelo de despliegue.

2.3.1 DIAGRAMA DE DESPLIEGUE.

El diseño se implementará siguiendo la distribución física común de una aplicación web, donde deben existir como mínimo dos nodos de procesamiento del sistema. A continuación se muestran cómo están distribuidos los nodos y dispositivos para este producto, en este caso están distribuidos en dos variantes.

Variante 1:

Instalar localmente el producto en la PC del usuario el cual va a tener instalado el servidor web y la base de datos. La impresora como dispositivo utilizado por el sistema para posibilitar la impresión de documentos.



Figura 2.1: Diagrama de Despliegue (variante1).

Variante 2:

Se necesita de un ordenador que funcione como servidor central, el cual tendrá instalado el servidor web y el servidor de base de datos. Además se necesita de una computadora que haga función de cliente, en el cual el usuario va a poder interactuar con la aplicación y la impresora como dispositivo utilizado por el sistema para posibilitar la impresión de documentos.

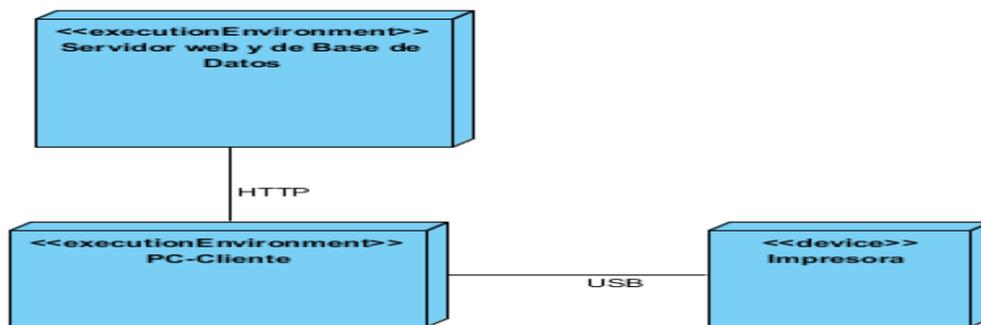


Figura 2.2: Diagrama de Despliegue (variante2).

Descripción de la funcionalidad y capacidad de los nodos.

PC Cliente: ordenador cliente que se conecta a través de un navegador web al servidor central donde reside la aplicación. No se necesita tener instalado el producto localmente.

PC Servidor: servidor central, el cual tiene instalado un servidor web, uno de base de datos y hospeda todos los componentes necesarios para el funcionamiento del producto.

Descripción de elementos e interfaces de comunicación.

<<HTTP>>

Representa la conexión que se va a establecer entre una PC Cliente con el servidor central donde reside el servidor web y el de base de datos, es decir, significa la conexión entre el navegador web y el servidor de aplicaciones.

<<USB>>

Conexión que existe entre la impresora y la PC Cliente del usuario. Se modela con USB, pero puede existir otro tipo, en dependencia del tipo de impresora que esté usando el usuario y el tipo de conexión que utilice esta.

2.3.2 DIAGRAMA DE COMPONENTES.

Los diagramas de componentes describen los elementos físicos del sistema y sus relaciones. Muestran las dependencias lógicas entre componentes software, sean estos componentes fuentes, binarios o ejecutables. Prevalecen en el campo de la arquitectura de software pero pueden ser usados para modelar y documentar cualquier arquitectura de sistema, ayudando a describir la vista de implementación estática de un sistema.

A continuación se muestra el Diagrama de Componentes general de la aplicación (*frontend*).

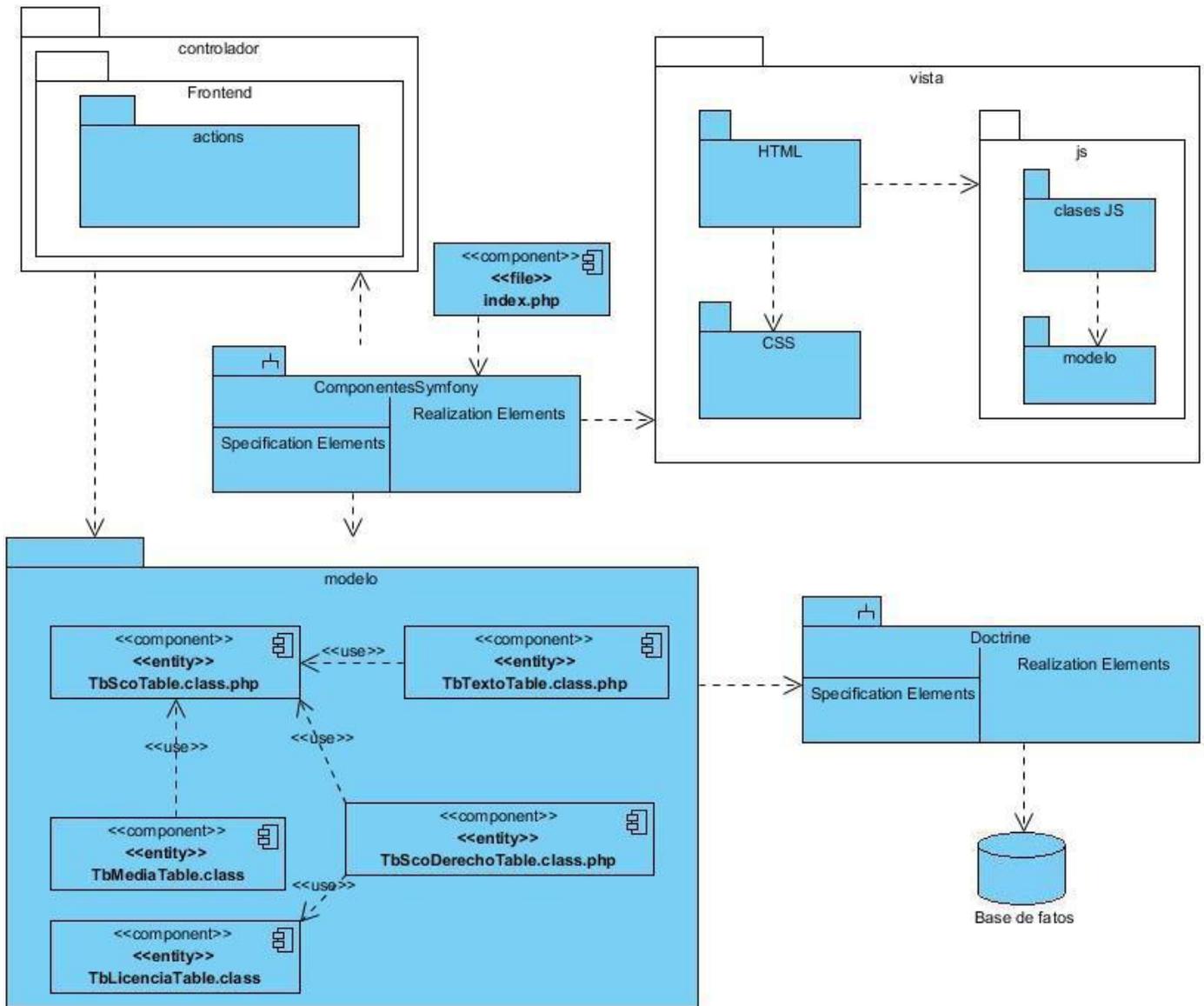


Figura 2.3: Diagrama de Componentes general del *frontend*.

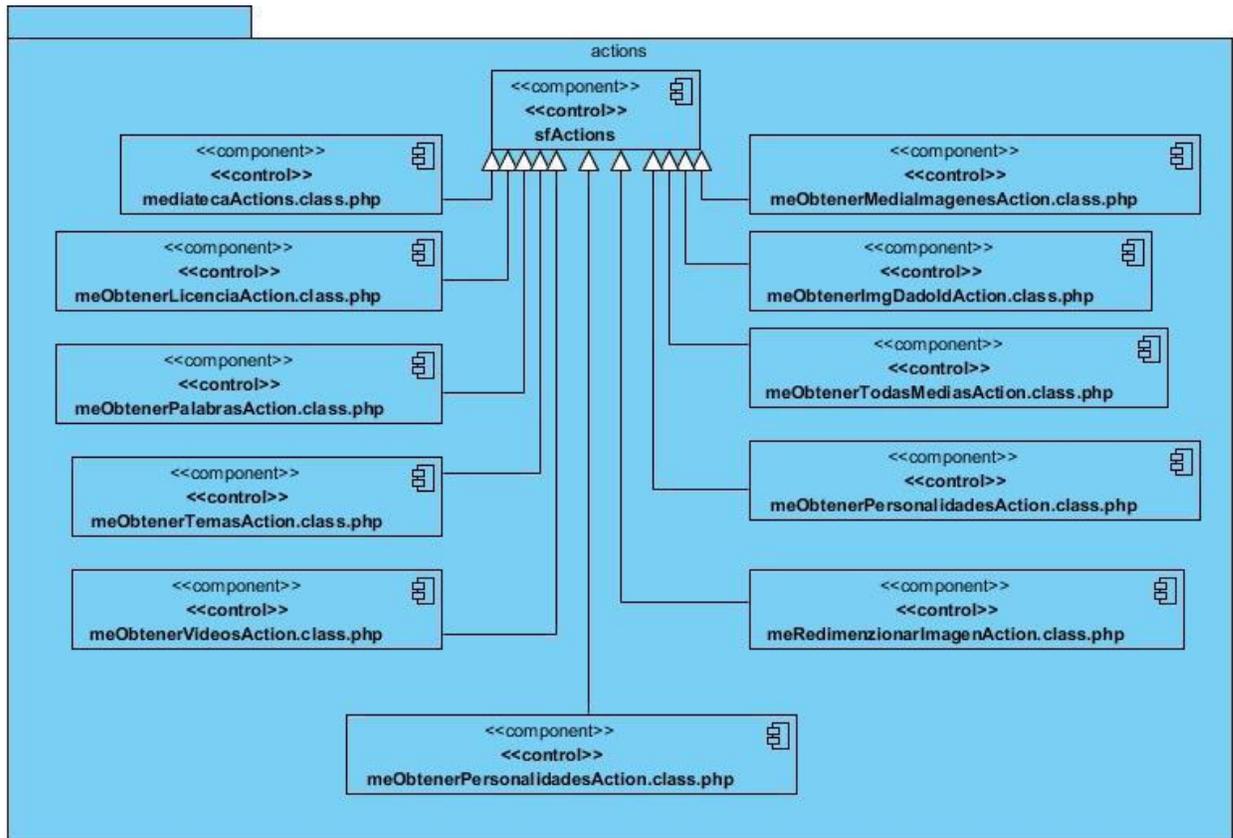


Figura 2.4: Diagrama de Componentes del paquete action del *frontend*.

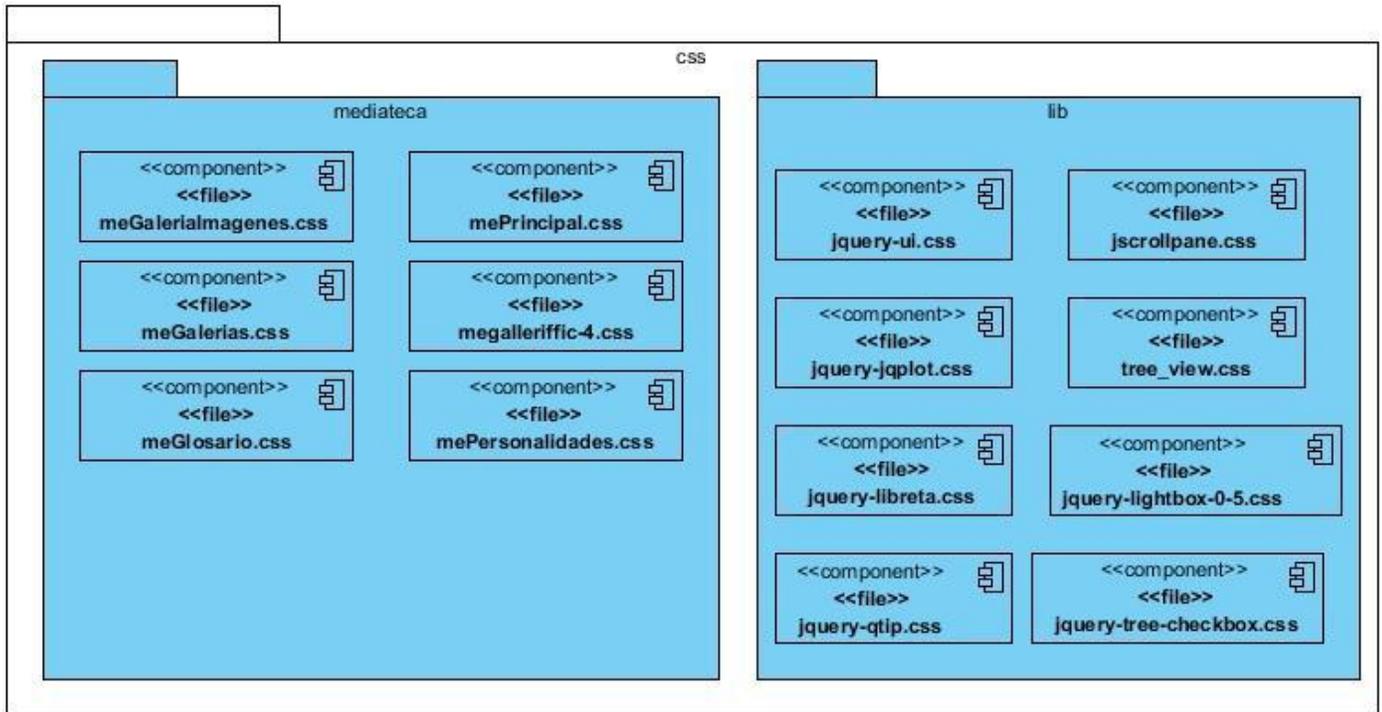


Figura 2.5: Diagrama de Componentes del paquete css del *frontend*.

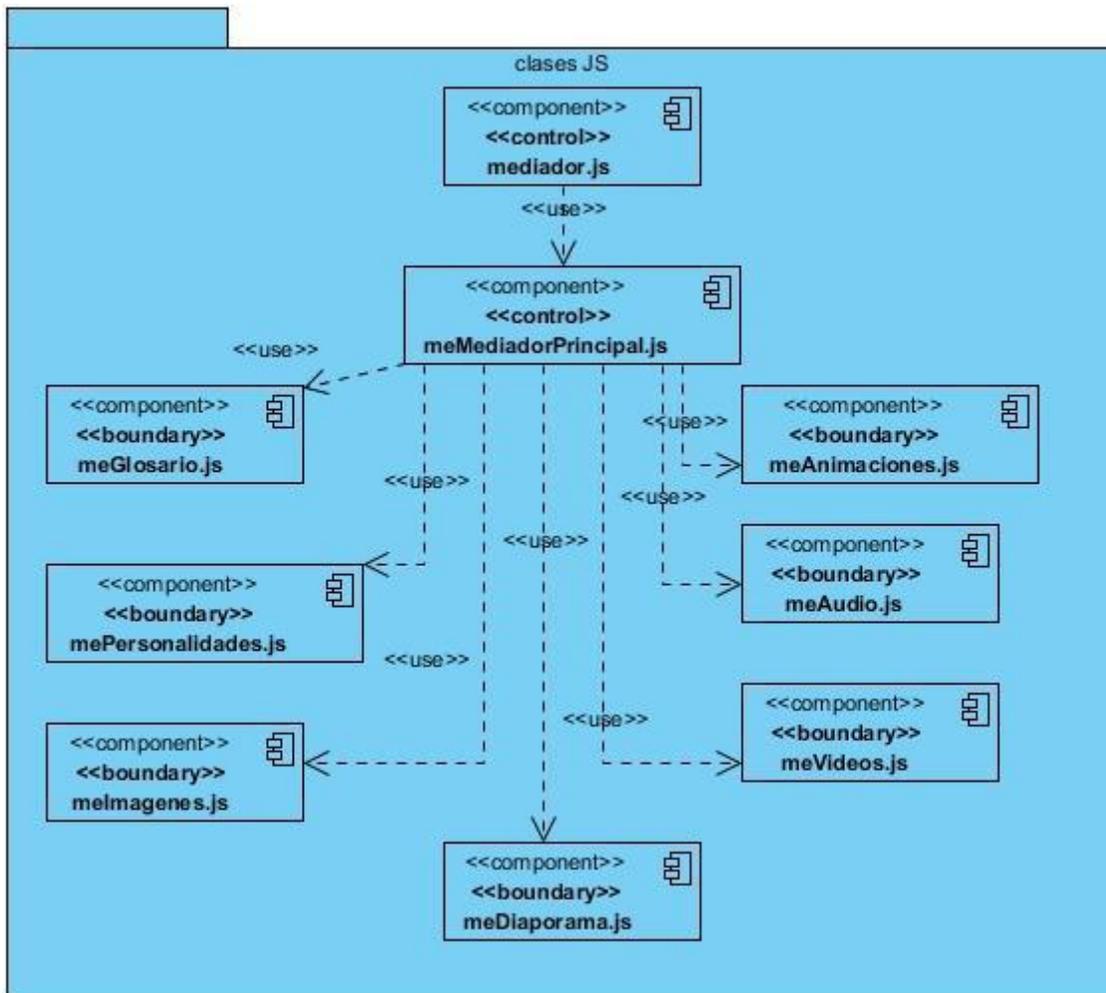


Figura 2.6: Diagrama de Componentes del paquete js del *frontend*.

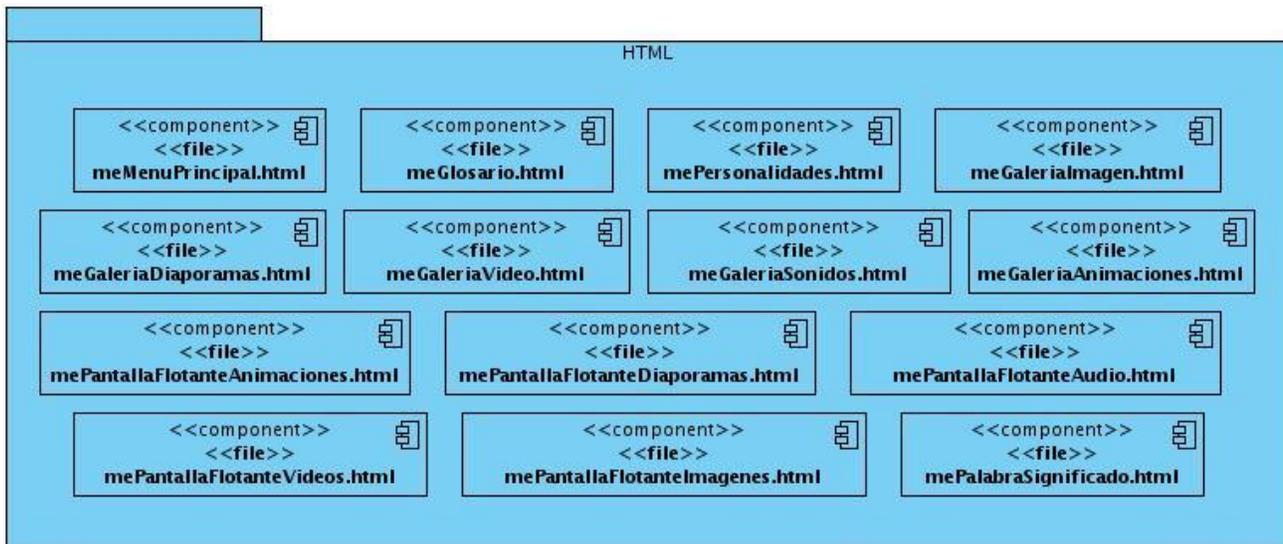


Figura 2.7: Diagrama de Componentes del paquete `html` del *frontend*.

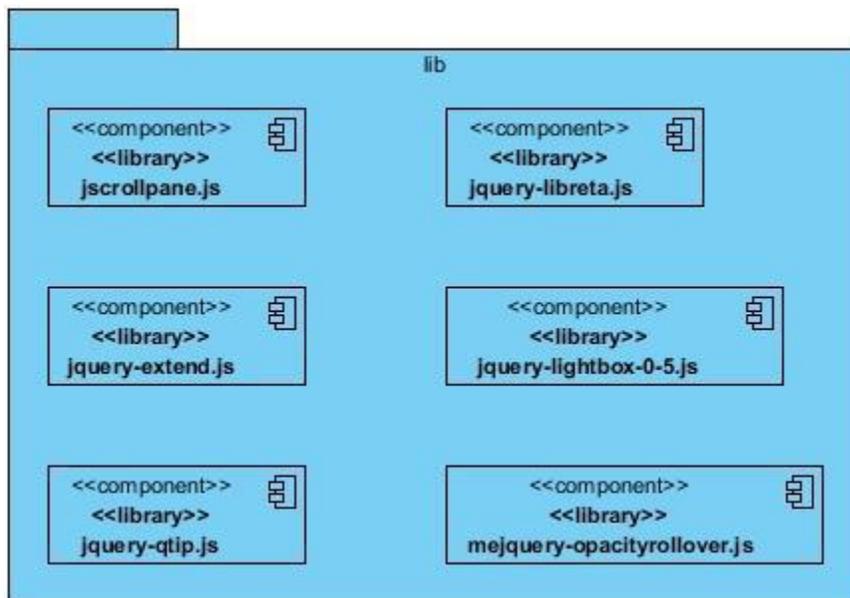


Figura 2.8: Diagrama de Componentes del paquete `lib` del *frontend*.

A continuación se muestra el Diagrama de Componentes general de la aplicación (*backend*).

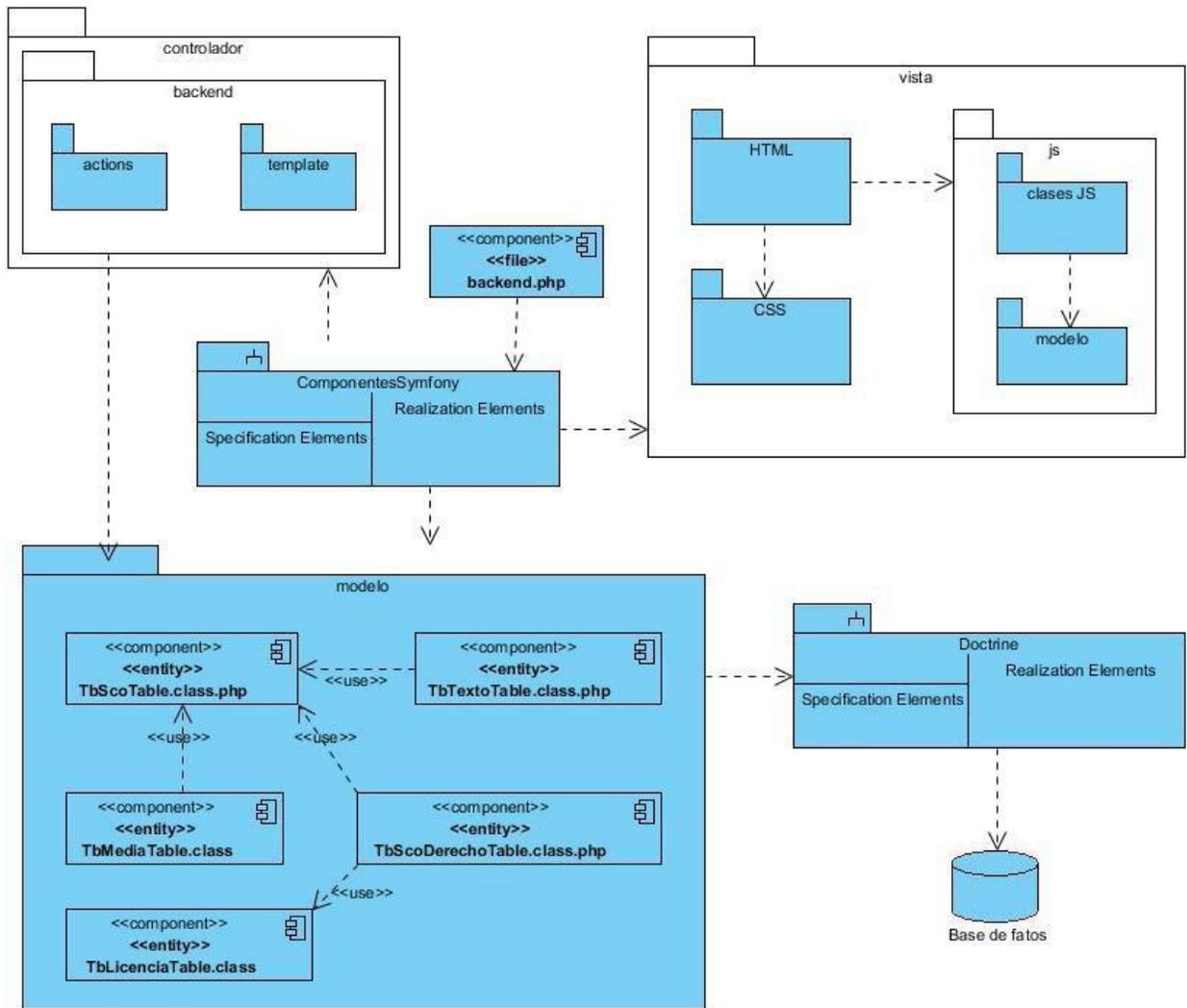


Figura 2.9: Diagrama de Componentes general del *backend*.

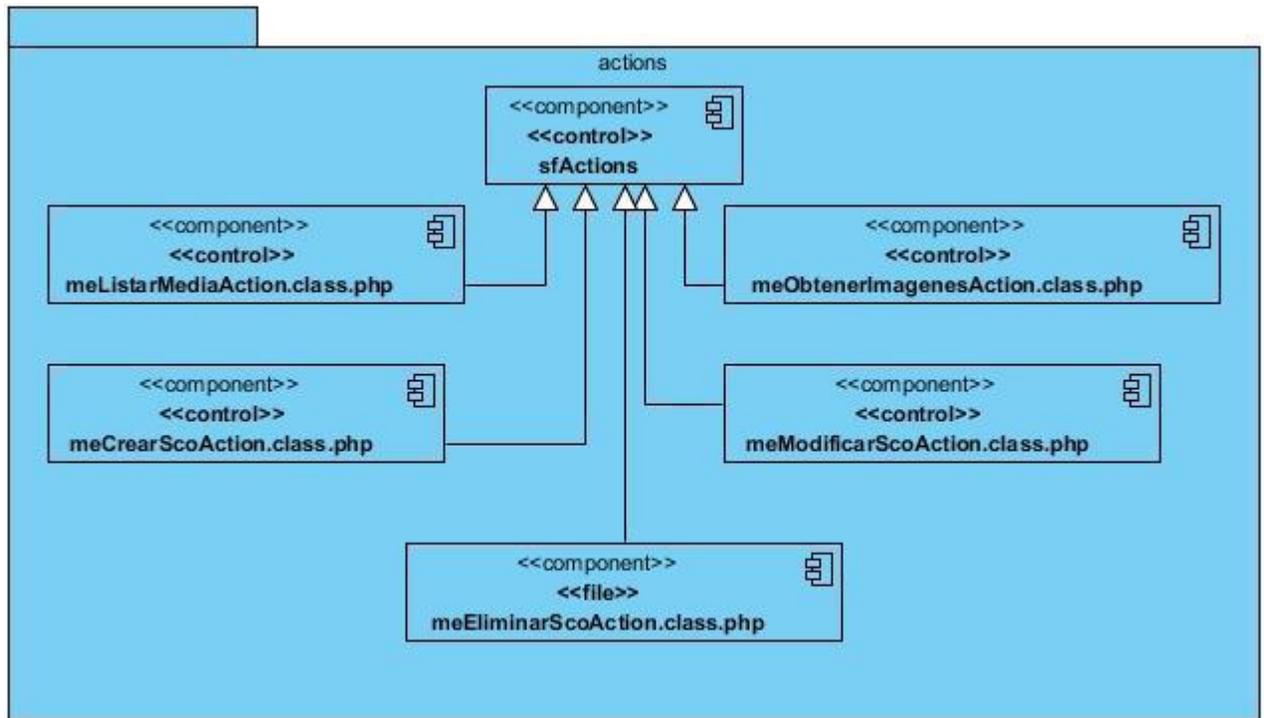


Figura 2.10: Diagrama de Componentes del paquete action del *backend*.

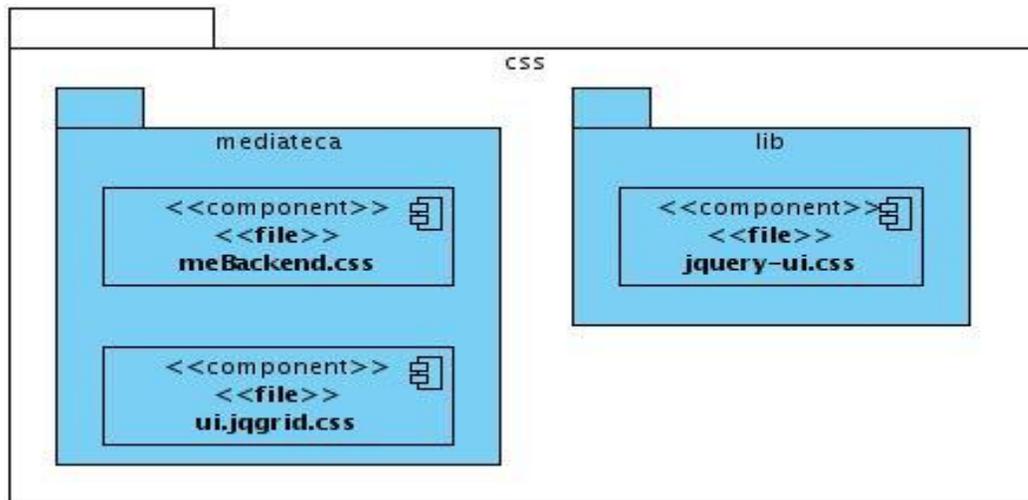


Figura 2.11: Diagrama de Componentes del paquete css del *backend*.

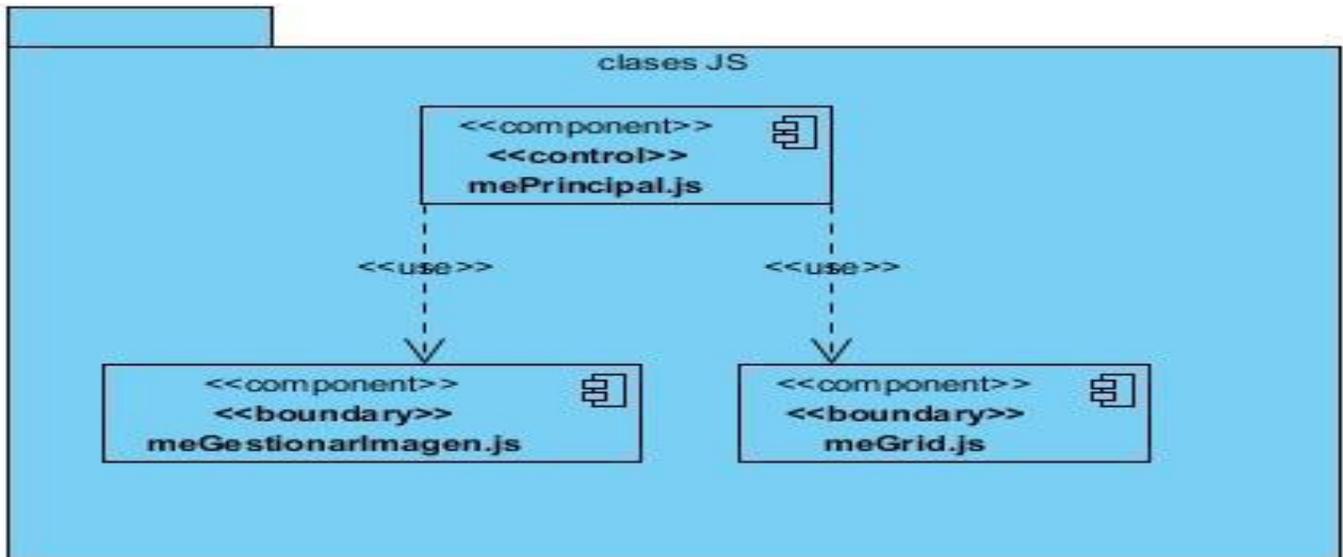


Figura 2.12: Diagrama de Componentes del paquete js del *backend*.



Figura 2.13: Diagrama de Componentes del paquete html del *backend*.



Figura 2.14: Diagrama de Componentes del paquete lib del *backend*.

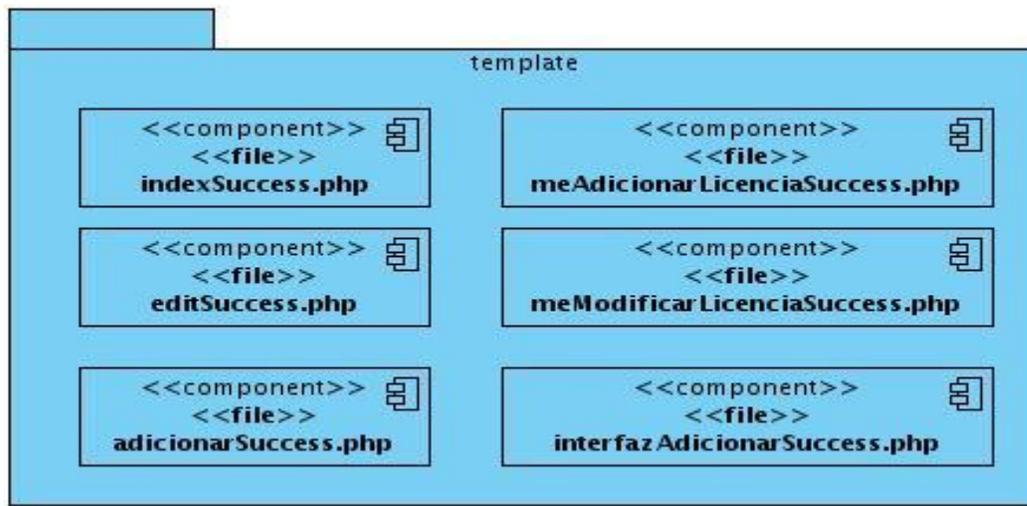


Figura 2.15: Diagrama de Componentes del paquete template del *backend*.

2.4 ESTÁNDAR DE CODIFICACIÓN.

Los estándares de codificación son reglas que se utilizan para la escritura del código fuente. Estas permiten asegurarse de que todos los programadores del proyecto trabajen de forma coordinada y con un vocabulario común y ayudan a crear un código más legible, reutilizable y de fácil mantenimiento.

2.4.1 COMENTARIOS.

Para garantizar un código más legible, reutilizable y que permita una adecuada comprensión para los otros programadores del proyecto, es que se hace necesario establecer este estándar de codificación. Se definieron 4 tipos de comentarios: bloques, líneas simples, líneas de seguimiento y fin de línea.

Bloques de líneas.

Un bloque de comentario es precedido por una línea en blanco para separarlo del resto del código. Los bloques de comentario tienen solo un asterisco al inicio de cada línea exceptuando la primera.

Ejemplo:

```
/**
```

*Se escribe aquí.

**/

Líneas simples.

Los comentarios cortos pueden aparecer en una sola línea, alineados al mismo nivel que el código que lo sigue.

Ejemplo:

```
if (true) {  
  
/* Condición principal*/  
  
}
```

Líneas de seguimiento.

Comentarios muy cortos pueden aparecer al final de la línea de código que describen, pero están alejados lo suficiente para separarlos de las sentencias.

```
if (true) {  
  
return valor(a); /* si se cumple*/  
  
} else {  
  
return valor(b); /* sino resultado*/  
  
}
```

Fin de línea.

El delimitador de comentario // convierte en comentario una línea completa o una parte de una línea.

2.4.2 IDENTACIÓN Y ESPACIOS EN BLANCO.

La indentación es usada para tener una mejor visibilidad en el diseño del código, por lo tanto, se tomará en cuenta a la hora de escribir cualquier fragmento de código, utilizándose siempre espacios en blanco y nunca tabulaciones, las estructuras anidadas usarán cuatro espacios para la indentación.

✓ **Alineación y longitud de líneas.**

Indentar con dos espacios como unidad de alineación y utilizar líneas entre 75 y 80 caracteres de longitud para que sean mejor manejadas por algunas herramientas y terminales.

✓ **Líneas plegadas.**

Cuando una expresión no cabe en una línea simple debido a su extensión se divide en más de una línea, siguiendo las siguientes precisiones:

- ✓ Dividir después de una coma.
- ✓ Dividir después de un operador.
- ✓ Alinear la nueva línea al inicio de la expresión en el mismo nivel que la línea anterior.

2.4.3 CONVENCIONES DE NOMBRE (FUNCIONES, VARIABLES GLOBALES, CLASES)

Los nombres que se utilizan son cortos, y de forma abreviada, siempre que su contexto sea específicamente local y su lectura sea intuitiva. Para los nombres se establecen las siguientes reglas:

Funciones y métodos

La notación **Camel** consiste en escribir los identificadores con la primera letra de cada palabra en mayúsculas y el resto en minúsculas: **Ej. IdMedia**.

Las funciones y métodos se nombran usando el formato camelCase (notación camello) que es una variante de la notación Camel donde la primera letra es minúscula. **Ej. meGrid()**.

Constantes

El código de Symfony no incluye ninguna constante, pues utiliza su propio objeto para almacenar la configuración, llamado **sfConfig**, y que reemplaza a las constantes.

Variables globales

Si se necesita definir variables globales, el nombre comenzará con guión bajo simple.

Clases

Los nombres de las clases comienzan con el **sufijo** del módulo, seguido del formato camelCase: Ej. **meMediadorPrincipal**. Las clases que genera Doctrine comienzan con:

Tb_: las clases de entidad.

nom_: las clases de nomencladores.

r_: las clases que representan relaciones muchos-a-muchos.

Atributos

Los atributos se nombran siguiendo el formato camelCase. Ej. **getData()**.

Los miembros privados de las clases (dígase aquellos que se usan solo en la misma clase donde son declarados; PHP aún no soporta espacios de nombres privados realmente obligatorios) comienzan con guión bajo. Ej. **\$_status, _sort(), _initTree()**

Nombre de las tablas en la base de datos

Los prefijos que se utilizan en la creación de tablas son los siguientes:

tb_ En tablas que almacenan la mayor cantidad de características de una entidad.

nom_ En tablas nomencladoras.

r_ Para establecer la relación de mucho a mucho entre tablas (cualquier tipo de tablas).

Acciones

Se utiliza una clase por acción, para esto el nombre de la clase debe ser nombreAcciónAction y el nombre del método es simplemente execute.

```
classlistarAction extends sfAction
```

```
{
```

```
public function execute($petición)
```

```
{
```

```
//...
```

```
}
```

```
}
```

2.5 CONCLUSIONES DEL CAPÍTULO.

En este capítulo se detallaron los artefactos fundamentales del flujo de implementación, mostrando los Diagramas de Componentes y de Despliegue, quedando confeccionada la vista estática del sistema. Se mostró una descripción de la estandarización de código y se mostraron los requisitos del sistema.

PRUEBA DE SOFTWARE DEL MÓDULO MEDiateca

3.1 INTRODUCCIÓN.

Las pruebas tienen una gran importancia en el desarrollo de software pues se consideran indispensables para obtener buenos resultados. Se realizan con el objetivo de revisar que todo el software tenga el nivel de calidad requerido. Permiten detectar errores durante el funcionamiento del software, probando así la entrada y salida correcta de datos. En la ejecución de las pruebas lo que se persigue es la detección de discrepancias entre la salida obtenida y la esperada. Dicha situación se interpreta entonces como un síntoma de problemas con el software. Los criterios de selección (denominados criterios de cobertura) suelen pretender la cobertura de un porcentaje o una representación del universo posible de elementos estructurales o de situaciones funcionales. (18)De hecho, algunos autores hablan más de selección de casos de pruebas que de diseño de los mismos. (19)

En este capítulo se definen y describen los casos de prueba para darle una mayor consistencia y confiabilidad al módulo desarrollado, utilizando el método de prueba de caja negra y la técnica partición equivalente, finalmente se muestran los resultados de las pruebas en un resumen de no conformidades.

3.2 MÉTODOS DE PRUEBA.

Los métodos se encargan de definir la estrategia de verificación y validación del sistema con el objetivo de demostrar que existen fallos, no la ausencia de estos. Tienen el objetivo de diseñar pruebas que descubran diferentes tipos de errores con menor tiempo y esfuerzo. Existen dos métodos de pruebas fundamentales: el método de caja negra (denominada prueba funcional) y el de caja blanca.

Pruebas de Caja Negra

Las pruebas de caja negra, también denominadas pruebas de comportamiento, se centran en los requisitos funcionales del software. O sea, permiten al ingeniero del software obtener conjuntos de condiciones de entrada que ejerciten completamente todos los requisitos funcionales de un programa. Estas pruebas no son una alternativa a las técnicas de pruebas de caja blanca, más bien se trata de un enfoque complementario que intenta descubrir diferentes tipos de errores que se llevan a cabo sobre la interfaz del software. El objetivo es demostrar que las funciones del software son operativas, que las entradas se aceptan de forma adecuada y se produce un resultado correcto, y que la integridad de la

información externa se mantiene, saber qué es lo que hace el software pero sin entrar en detalles de código, es decir, qué es lo que hace, y no cómo lo hace (no se ve el código). Estas pruebas se centran principalmente en los requisitos funcionales del software y permiten encontrar: (20)

Funciones incorrectas o ausentes.

- ✓ Errores de interfaz.
- ✓ Errores en estructuras de datos o en accesos a las bases de datos externas.
- ✓ Errores de rendimiento.
- ✓ Errores de inicialización y terminación.

Para la realización de las pruebas de cajas negras, se utilizan varias técnicas, dentro de ellas se encuentran: Técnica de la Partición de Equivalencia, Análisis de Valores Límites y Grafos de Causa-Efecto. La técnica que se seleccionó para realizar los casos de prueba es la variante de particiones equivalentes, cuyo objetivo fundamental es dividir el dominio de entrada de un programa en clases de datos de los que se pueden derivar casos de prueba. Esta técnica consta de dos pasos fundamentales: (20)

- ✓ Identificación de las clases de equivalencia.
- ✓ Identificación de casos de prueba.

En el primer paso, las clases de equivalencia son identificadas tomando cada condición de entrada (generalmente una oración o una frase en la especificación) y repartiéndola en dos o más grupos, cada combinación será un escenario del caso de uso. En el segundo paso se identifican las variables y las clases de equivalencia para la confección de los casos de prueba.

Pruebas de Caja Blanca

La prueba de caja blanca, denominada a veces prueba de caja de cristal es un método de diseño de casos de prueba que usa la estructura de control del diseño procedimental para obtener los casos de prueba. Mediante los métodos de prueba de caja blanca, el ingeniero del software puede obtener casos de prueba que garanticen que se ejercita por lo menos una vez todos los caminos independientes de cada

módulo, ejerciten todas las decisiones lógicas en sus vertientes verdadera y falsa; ejecuten todos los ciclos en sus límites y con sus límites operacionales, y ejerciten las estructuras internas de datos para asegurar su validez. (21)

Partiendo de lo planteado se decide realizar pruebas de cajas negras con el objetivo de validar que la versión 1.1 del módulo Mediateca cumple con los requerimientos definidos por el cliente.

3.3 CLASIFICACIÓN DE LAS NO CONFORMIDADES.

La descripción de las clasificaciones de las no conformidades (NC) ayudará a la acelerada gestión de los proyectos y a que el proceso de pruebas de liberación se realice en un tiempo reducido y planificado, sin atender contra el cronograma de desarrollo y los compromisos con el cliente. Se podrá realizar una evaluación más profunda del software a revisar, se tendrán mejores estadísticas de cuáles son las no conformidades más comunes en los diferentes tipos de software y se le podrá dar un mejor tratamiento a la no conformidad en el proceso de liberación del software. (22)

Una no conformidad según lo planteado en la Norma UNE-EN ISO 9001:2000. “Es el incumplimiento de algunos de los requisitos”. (22)

Las mismas se clasifican de acuerdo al nivel de importancia en:

- ✓ Las NC significativas son aquellas que afectan la calidad del producto o servicio de manera visible, impidiendo o no el cumplimiento de algún requisito.
- ✓ Las NC no significativas resultan aquellas menos visibles que no atentan el cumplimiento de algún requisito.
- ✓ Las clasificadas como recomendaciones quedan en función de la apreciación del probador de oportunidades de mejoras del producto o servicio.

Además de las clasificaciones antes mencionadas se formulan otras, atendiendo a las características que presenta la NC.

No conformidades de documentación:

- ✓ Formato.

- ✓ Error técnico.
- ✓ Otros errores.
- ✓ Correspondencia con otra documentación.

No conformidades de aplicación:

- ✓ Validación.
- ✓ Opciones que no funcionan.
- ✓ Errores de interfaz.
- ✓ Funcionalidad.
- ✓ Excepciones.
- ✓ Correspondencia de lo implementado con lo documentado.

No conformidades comunes (para ambos artefactos):

- ✓ Ortografía.
- ✓ Redacción.
- ✓ Errores de idioma.

3.4 DISEÑO DE CASOS DE PRUEBA.

A continuación se presenta el diseño de casos de prueba para el **Caso de Uso (CU) Gestionar Imagen**.

Descripción General.

El caso de uso se inicia cuando el administrador del sistema selecciona la opción Imágenes para gestionar. El sistema muestra el listado de las imágenes existentes y permite adicionar una nueva imagen, modificar los datos de la imagen seleccionada, eliminarla y brinda la posibilidad de filtrar las imágenes de acuerdo a un conjunto de criterios de búsqueda. El administrador selecciona una de estas opciones y finaliza el caso de uso.

Condiciones de Ejecución.

Debe haberse generado el escritorio de trabajo del usuario autenticado.

Secciones a probar en el caso de uso

Nombre de la sección	Escenarios de la sección	Descripción de la funcionalidad
SC 1: Adicionar imagen.	EC 1.1: Adicionar una imagen.	El usuario inserta todos los datos de la imagen.
	EC1.2: Existen datos incompletos.	Muestra una señalización sobre los campos que debe llenar. Muestra una señalización sobre el formulario indicando que debe llenar los campos requeridos.
	EC1.3: Introduce código de imagen existente.	Muestra una señalización sobre el campo código. Muestra una señalización sobre el formulario indicando que debe llenar los campos requeridos.
SC 2: Modificar imagen.	EC 2.1: Modificar datos de la imagen.	El usuario modifica los datos de la imagen.
	EC 2.2: Existen datos incompletos.	Muestra una señalización sobre los campos que debe llenar.
	EC 2.3: La extensión de la imagen no es válida.	Muestra una señalización sobre el formulario indicando que tipo de extensión es válida para las imágenes.
	EC 2.4: Regresar al listado.	Muestra un listado con las imágenes existentes.
SC 3: Eliminar imagen.	EC 3.1: Eliminar imagen.	Está seguro que desea eliminar la media seleccionada.
	EC 3.2. Se selecciona Aceptar.	Se elimina la imagen deseada.

	EC 3.3. Se selecciona Cancelar.	Se muestra el listado de las imágenes.
--	---------------------------------	--

Tabla1: Secciones a probar en el CU Gestionar Imagen.

Descripción de las variables

No	Nombre de campo	Clasificación	Valor Nulo	Descripción
1	Nombre	Campo de texto	No	Nombre de la media.
2	Descripción	Campo de texto	No	Descripción de la media.
3	Temas asociados	Campo de selección	Si	Temas asociados a la media.
4	Licencias	Lista desplegable	Si	Licencia de la media.
5	Autor	Campo de texto	No	Gestor de contenido
6	Visible	Campo booleano.	Si	Indica si la media se ve en la mediateca.
7	Media	Campo de texto	No	Url de la media que se va a adicionar.
8	Descripción tema	Campo de texto	Si	Descripción de la media para el módulo temas.
9	Código	Campo de texto	No	Código de la media.

Tabla2: Descripción de las variables para el caso de prueba en el CU Gestionar Imagen.

Matriz de datos

SC #1: Adicionar imagen

ID del escenario	Escenario	Nombre	Descripción	Temas Asociados	Licencias	Autor	Visible	Media	Descripción Tema	Código	Respuesta sistema	Flujo central

EC 1	Selecciona la opción adicionar imagen.										Brinda la posibilidad de Insertar los datos de una imagen especificando: <ul style="list-style-type: none"> • Nombre • Descripción • Temas Asociados • Licencias • Autor • Visible • Media • Descripción Tema • Código 	Pantalla principal del Navegante(backend) /Módulo Mediateca/Interfaz Adicionar Media.
EC 1.1	Insertar datos de la imagen	V	V	V	V	V	V	V	N/A	V	Valida los datos. Inserta los datos. Muestra un listado de las imágenes.	Pantalla principal del Navegante(backend) /módulo Mediateca/ Interfaz Adicionar Media/Botón Salvar/
EC 1.2	Existen datos incompletos.	I	V	V	V	V	V	V	N/A	V	Muestra una señalización sobre los campos que debe llenar. Muestra una señalización sobre el formulario indicando que debe llenar los campos requeridos. Regresa al EC 1.1	Pantalla principal del Navegante(backend) /módulo Mediateca/
		V	I	V	V	V	V	V	N/A	V		
		V	V	I	V	V	V	V	N/A	V		
		V	V	V	I	V	V	V	N/A	V		
		V	V	V	V	I	V	V	N/A	V		
		V	V	V	V	V	V	I	N/A	V		
		V	V	V	V	V	V	V	I	V		
		V	V	V	V	V	V	V	N/A	I		
EC 1.3	Introduce código de imagen existente.	V	V	V	V	V	V	V	N/A	I	Muestra una señalización sobre el campo código. Muestra una señalización sobre el formulario indicando que debe llenar los campos requeridos. Regresa al EC 1.1	Pantalla principal del Navegante(backend) /módulo Mediateca/

Tabla3: Matriz de datos caso de prueba del CU Gestionar Imagen SC Adicionar imagen.

SC #2: Modificar imagen

ID del escenario	Escenario	Nombre	Descripción	Temas Asociados	Licencias	Autor	Visible	Media	Descripción Tema	Código	Respuesta sistema	Flujo central
EC 1	Selecciona la opción modificar imagen.										Brinda la posibilidad de Editar los datos previamente guardados de la imagen : <ul style="list-style-type: none"> • Nombre • Descripción • Temas Asociados • Licencias • Autor • Visible • Media • Descripción Tema • Código 	Pantalla principal del Navegante(backend) /módulo Mediateca/Interfaz Modificar Media.
EC 1.1	Modificar los datos de la imagen.	V	V	V	V	V	V	V	N/A	V	Valida los datos. Actualiza los datos. Muestra una señalización sobre el formulario indicando que ha modificado correctamente.	Pantalla principal del Navegante(backend) /módulo Mediateca/Interfaz Modificar Media/Botón Salvar/
EC 1.2	Existen datos incompletos.	I	V	V	V	V	V	V	N/A	V	Muestra una señalización sobre los campos que debe llenar. Muestra una señalización sobre el formulario indicando que debe llenar los campos requeridos. Regresa al EC 1.1	Pantalla principal del Navegante(backend) /módulo Mediateca/
		V	I	V	V	V	V	V	N/A	V		
		V	V	I	V	V	V	V	N/A	V		
		V	V	V	I	V	V	V	N/A	V		
		V	V	V	V	I	V	V	N/A	V		
		V	V	V	V	V	V	I	N/A	V		
		V	V	V	V	V	V	V	I	V		
		V	V	V	V	V	V	V	N/A	I		

EC 1.3	La extensión de la imagen no es válida.	V	V	V	V	V	V	I	N/A	V	Muestra una señalización sobre el formulario indicando que tipo de extensión es válida para las imágenes. Regresa al EC 1.1	Pantalla principal del Navegante(backend) /módulo Mediateca/
EC 1.4	Regresar al listado.										Muestra un listado con las imágenes existentes.	Pantalla principal del Navegante(backend) /módulo Mediateca/

Tabla4: Matriz de datos caso de prueba del CU Gestionar Imagen SC Modificar imagen.

SC #3: Eliminar imagen

Escenario	Descripción	Respuesta del sistema	Flujo central
EC 1	Selecciona la opción eliminar imagen.	Muestra un mensaje de advertencia para confirmar la eliminación de la imagen y permite seleccionar las opciones. <ul style="list-style-type: none"> • Aceptar • Cancelar 	Pantalla principal del Navegante(backend) /módulo Mediateca/
EC 1.1	Se selecciona la opción Aceptar.	Elimina la imagen y muestra un mensaje indicando que la imagen seleccionada ha sido eliminada correctamente.	Pantalla principal del Navegante(backend) /módulo Mediateca/Botón Aceptar
EC 1.3	Se selecciona la opción Cancelar.	Muestra el listado de las Imágenes.	Pantalla principal del Navegante(backend) /módulo Mediateca/Botón Cancelar

Tabla5: Matriz de datos caso de prueba del CU Gestionar Imagen SC Eliminar imagen.

A continuación se presenta el diseño de casos de prueba para el **CU Gestionar Video**.

Descripción General

El caso de uso se inicia cuando el administrador del sistema selecciona la opción Video para gestionar. El sistema muestra el listado de los videos existentes y permite adicionar un nuevo video, modificar los datos de un video seleccionado, eliminarlo y brinda la posibilidad de filtrar los videos de acuerdo a un conjunto de criterios de búsqueda. El administrador selecciona una de estas opciones y finaliza el caso de uso.

Condiciones de Ejecución

Debe haberse generado el escritorio de trabajo del usuario autenticado.

Secciones a probar en el caso de uso

Nombre de la sección	Escenarios de la sección	Descripción de la funcionalidad
SC 1: Adicionar video.	EC 1.1: Adicionar un video.	El usuario inserta todos los datos de un video.
	EC1.2: Existen datos incompletos.	Muestra una señalización sobre los campos que debe llenar. Muestra una señalización sobre el formulario indicando que debe llenar los campos requeridos.
	EC1.3: Introduce código de un video existente.	Muestra una señalización sobre el campo código. Muestra una señalización sobre el formulario indicando que debe llenar los campos requeridos.
SC 2: Modificar Video.	EC 2.1: Modificar datos de un video.	El usuario modifica los datos de un video.
	EC 2.2: Existen datos incompletos.	Muestra una señalización sobre los campos que debe llenar.
	EC 2.3: La extensión de la imagen asociada no es válida.	Muestra una señalización sobre el formulario indicando que tipo de extensión es válida para la imagen asociada.

	EC 2.4: La extensión del video no es válida.	Muestra una señalización sobre el formulario indicando que tipo de extensión es válida para el video.
	EC 2.5: Regresar al listado.	Muestra un listado con los videos existentes.
SC 3: Eliminar video.	EC 3.1: Eliminar video.	Está seguro que desea eliminar la media seleccionada.
	EC 3.2. Se selecciona Aceptar.	Se elimina el video deseado.
	EC 3.3. Se selecciona Cancelar.	Se muestra el listado de los videos.

Tabla6: Secciones a probar en el CU Gestionar Video.

Descripción de las variables

No	Nombre de campo	Clasificación	Valor Nulo	Descripción
1	Nombre	Campo de texto	No	Nombre de la media.
2	Descripción	Campo de texto	No	Descripción de la media.
3	Temas asociados	Campo de selección	Si	Temas asociados a la media.
4	Licencias	Lista desplegable	Si	Licencia de la media.
5	Autor	Campo de texto	No	Gestor de contenido
6	Imagen asociada	Campo de texto	Si	Url de la imagen asociada que se va adicionar.
7	Visible	Campo booleano.	Si	Indica si la media se ve en la mediateca.

8	Media	Campo de texto	No	Url de la media que se va a adicionar.
9	Descripción tema	Campo de texto	Si	Descripción de la media para el módulo temas.
10	Código	Campo de texto	No	Código de la media.

Tabla 7: Descripción de las variables para el caso de prueba en el CU Gestionar Video.

Matriz de datos

SC #1: Adicionar video

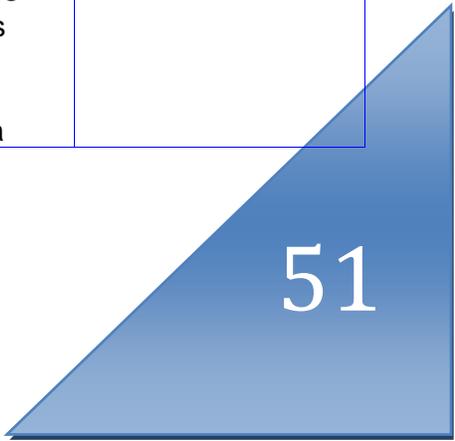
ID del escenario	Escenario	Nombre	Descripción	Temas Asociados	Licencias	Autor	Imagen Asociada	Visible	Media	Descripción Tema	Código	Respuesta sistema	Flujo central
EC 1	Selecciona la opción adicionar video.											Brinda la posibilidad de Insertar los datos de un video especificando: <ul style="list-style-type: none"> • Nombre • Descripción • Temas Asociados • Licencias • Autor • Imagen Asociada • Visible • Media • Descripción Tema • Código 	Pantalla principal del Navegante(backend) /módulo Mediateca/Interfaz Adicionar Media.
EC 1.1	Insertar datos de un video	V	V	V	V	V	V	V	V	N/A	V	Valida los datos. Inserta los datos. Muestra un listado de los videos.	Pantalla principal del Navegante(backend) /módulo Mediateca/Interfaz Adicionar Media/Botón Salvar/
EC 1.2	Existen datos											Muestra una	Pantalla principal del

	incompletos.	I	V	V	V	V	V	V	V	N/A	V	señalización sobre los campos que debe llenar. Muestra una señalización sobre el formulario indicando que debe llenar los campos requeridos. Regresa al EC 1.1	Navegante(backend)/módulo Mediateca/
		V	I	V	V	V	V	V	V	N/A	V		
		V	V	I	V	V	V	V	V	N/A	V		
		V	V	V	I	V	V	V	V	N/A	V		
		V	V	V	V	I	V	V	V	N/A	V		
		V	V	V	V	V	I	V	V	N/A	V		
		V	V	V	V	V	V	I	N/A	V			
		V	V	V	V	V	V	V	I	V			
EC 1.3	Introduce código de un video existente.	V	V	V	V	V	V	V	V	N/A	I	Muestra una señalización sobre el campo código. Muestra una señalización sobre el formulario indicando que debe llenar los campos requeridos. Regresa al EC 1.1	Pantalla principal del Navegante(backend)/módulo Mediateca/

Tabla 8: Matriz de datos caso de prueba del CU Gestionar Video SC Adicionar video.

SC #2: Modificar video

ID del escenario	Escenario	Nombre	Descripción	Temas Asociados	Licencias	Autor	Imagen Asociada	Visible	Media	Descripción Tema	Código	Respuesta sistema	Flujo central
EC 1	Selecciona la opción modificar video.											Brinda la posibilidad de editar los datos previamente guardados del video: <ul style="list-style-type: none"> • Nombre • Descripción • Temas Asociados • Licencias • Autor • Imagen Asociada 	Pantalla principal del Navegante(backend)/módulo Mediateca/Interfaz Modificar Media.



												<ul style="list-style-type: none"> • Visible • Media • Descripción Tema • Código 	
EC 1.1	Modificar los datos de un video.	V	V	V	V	V	V	V	V	N/A	V	Valida los datos. Actualiza los datos. Muestra una señalización sobre el formulario indicando que ha Modificado correctamente.	Pantalla principal del Navegante(backend) /módulo Mediateca/ Interfaz Modificar Media/Botón Salvar/
EC 1.2	Existen datos incompletos.	I	V	V	V	V	V	V	V	N/A	V	Muestra una señalización sobre los campos que debe llenar. Muestra una señalización sobre el formulario indicando que debe llenar los campos requeridos.	Pantalla principal del Navegante(backend) /módulo Mediateca/
		V	I	V	V	V	V	V	V	N/A	V		
		V	V	I	V	V	V	V	V	N/A	V		
		V	V	V	I	V	V	V	V	N/A	V		
		V	V	V	V	I	V	V	V	N/A	V		
		V	V	V	V	V	V	I	V	N/A	V		
		V	V	V	V	V	V	V	V	N/A	V	Regresa al EC 1.1	
		V	V	V	V	V	V	V	V	N/A	I		
EC 1.3	La extensión de la imagen asociada no es válida.	V	V	V	V	V	I	V	V	N/A	V	Muestra una señalización sobre el formulario indicando que tipo de extensión es válida para las imágenes. Regresa al EC 1.1	Pantalla principal del Navegante(backend) /módulo Mediateca/

EC 1.4	La extensión del video no es válida.	V	V	V	V	V	V	V	I	N/A	V	Muestra una señalización sobre el formulario indicando que tipo de extensión es válida para los videos. Regresa al EC 1.1	Pantalla principal del Navegante(backend) /módulo Mediateca/
EC 1.5	Regresar al listado.											Muestra un listado con los videos existentes.	Pantalla principal del Navegante(backend) /módulo Mediateca/

Tabla 9: Matriz de datos caso de prueba del CU Gestionar Video SC Modificar video.

SC #3: Eliminar video

Escenario	Descripción	Respuesta del sistema	Flujo central
EC 1	Selecciona la opción eliminar video.	Muestra un mensaje de advertencia para confirmar la eliminación del video y permite seleccionar las opciones. <ul style="list-style-type: none"> • Aceptar • Cancelar 	Pantalla principal del Navegante(backend) /módulo Mediateca/
EC 1.1	Se selecciona la opción Aceptar.	Elimina el video y muestra un mensaje indicando que el video seleccionado ha sido eliminado correctamente.	Pantalla principal del Navegante(backend) /módulo Mediateca/Botón Aceptar
EC 1.3	Se selecciona la opción Cancelar.	Muestra el listado de los videos.	Pantalla principal del Navegante(backend) /módulo Mediateca/Botón Cancelar

Tabla 10: Matriz de datos caso de prueba del CU Gestionar Video SC Eliminar video.

A continuación se presenta el diseño de casos de prueba para el **CU Gestionar Sonido**.

Descripción General

El caso de uso se inicia cuando el administrador del sistema selecciona la opción Sonido para gestionar. El sistema muestra el listado de los sonidos existentes y permite adicionar un nuevo sonido, modificar los datos de un sonido seleccionado, eliminarlo y brinda la posibilidad de filtrar los sonidos de acuerdo a un conjunto de criterios de búsqueda. El administrador selecciona una de estas opciones y finaliza el caso de uso.

Condiciones de Ejecución

Debe haberse generado el escritorio de trabajo del usuario autenticado.

Secciones a probar en el caso de uso

Nombre de la sección	Escenarios de la sección	Descripción de la funcionalidad
SC 1: Adicionar sonido.	EC 1.1: Adicionar un sonido.	El usuario inserta todos los datos de un sonido.
	EC1.2: Existen datos incompletos.	Muestra una señalización sobre los campos que debe llenar. Muestra una señalización sobre el formulario indicando que debe llenar los campos requeridos.
	EC1.3: Introduce código de un sonido existente.	Muestra una señalización sobre el campo código. Muestra una señalización sobre el formulario indicando que debe llenar los campos requeridos.
SC 2: Modificar sonido.	EC 2.1: Modificar datos de un sonido.	El usuario modifica los datos de un sonido.
	EC 2.2: Existen datos incompletos.	Muestra una señalización sobre los campos que debe llenar.

	EC 2.3: La extensión de la imagen asociada no es válida.	Muestra una señalización sobre el formulario indicando que tipo de extensión es válida para la imagen asociada.
	EC 2.4: La extensión del sonido no es válida.	Muestra una señalización sobre el formulario indicando que tipo de extensión es válida para el sonido.
	EC 2.5: Regresar al listado.	Muestra un listado con los sonidos existentes.
SC 3: Eliminar sonido.	EC 3.1: Eliminar sonido.	Está seguro que desea eliminar la media seleccionada.
	EC 3.2. Se selecciona Aceptar.	Se elimina el sonido deseado.
	EC 3.3. Se selecciona Cancelar.	Se muestra el listado de los sonidos.

Tabla11: Secciones a probar en el CU Gestionar Sonido.

Descripción de las variables

No	Nombre de campo	Clasificación	Valor Nulo	Descripción
1	Nombre	Campo de texto	No	Nombre de la media.
2	Descripción	Campo de texto	No	Descripción de la media.
3	Temas asociados	Campo de selección	Si	Temas asociados a la media.
4	Licencias	Lista desplegable	Si	Licencia de la media.
5	Autor	Campo de texto	No	Gestor de contenido

6	Imagen asociada	Campo de texto	Si	Url de la imagen asociada que se va adicionar.
7	Visible	Campo booleano.	Si	Indica si la media se ve en la mediateca.
8	Media	Campo de texto	No	Url de la media que se va adicionar.
9	Descripción tema	Campo de texto	Si	Descripción de la media para el módulo temas.
10	Código	Campo de texto	No	Código de la media.

Tabla 12: Descripción de las variables para el caso de prueba en el CU Gestionar Sonido.

Matriz de datos

SC #1: Adicionar sonido

ID del escenario	Escenario	Nombre	Descripción	Temas Asociados	Licencias	Autor	Imagen Asociada	Visible	Media	Descripción Tema	Código	Respuesta sistema	Flujo central
EC 1	Selecciona la opción adicionar sonido.											Brinda la posibilidad de insertar los datos de un sonido especificando: <ul style="list-style-type: none"> • Nombre • Descripción • Temas Asociados • Licencias • Autor • Imagen Asociada • Visible • Media • Descripción Tema • Código 	Pantalla principal del Navegante(backend) /módulo Mediateca/Interfaz Adicionar Media.
EC 1.1	Insertar datos de un sonido	V	V	V	V	V	V	V	V	N/A	V	Valida los datos. Inserta los datos. Muestra un listado de	Pantalla principal del Navegante(backend) /módulo Mediateca/

												los sonidos.	Interfaz Adicionar Media/Botón Salvar/
EC 1.2	Existen datos incompletos.	I	V	V	V	V	V	V	V	N/A	V	Muestra una señalización sobre los campos que debe llenar. Muestra una señalización sobre el formulario indicando que debe llenar los campos requeridos. Regresa al EC 1.1	Pantalla principal del Navegante(backend)/módulo Mediateca/
		V	V	V	V	V	V	V	V	N/A	I		
EC 1.3	Introduce código de un sonido existente.	V	V	V	V	V	V	V	V	N/A	I	Muestra una señalización sobre el campo código. Muestra una señalización sobre el formulario indicando que debe llenar los campos requeridos. Regresa al EC 1.1	Pantalla principal del Navegante(backend)/módulo Mediateca/

Tabla 13: Matriz de datos caso de prueba del CU Gestionar Sonido SC Adicionar sonido.

SC #2: Modificar sonido

ID del escenario	Escenario	Nombre	Descripción	Temas Asociados	Licencias	Autor	Imagen Asociada	Visible	Media	Descripción Tema	Código	Respuesta sistema	Flujo central del sistema
EC 1	Selecciona la opción modificar sonido.											Brinda la posibilidad de editar los datos previamente guardados del sonido: <ul style="list-style-type: none"> • Nombre • Descripción 	Pantalla principal del Navegante(backend)/módulo Mediateca/Interfaz Modificar Media.

												<ul style="list-style-type: none"> • Temas Asociados • Licencias • Autor • Imagen Asociada • Visible • Media • Descripción Tema • Código 	
EC 1.1	Modificar los datos de un sonido.	V	V	V	V	V	V	V	V	N/A	V	<p>Valida los datos. Actualiza los datos. Muestra una señalización sobre el formulario indicando que ha modificado correctamente.</p>	Pantalla principal del Navegante(backend) /módulo Mediateca/ Interfaz Modificar Media/Botón Salvar/
EC 1.2	Existen datos incompletos.	I	V	V	V	V	V	V	V	N/A	V	Muestra una señalización sobre los campos que debe llenar. Muestra una señalización sobre el formulario indicando que debe llenar los campos requeridos.	Pantalla principal del Navegante(backend) /módulo Mediateca/
		V	I	V	V	V	V	V	V	N/A	V	Regresa al EC 1.1	
		V	V	I	V	V	V	V	V	N/A	V		
		V	V	V	I	V	V	V	V	N/A	V		
		V	V	V	V	V	I	V	V	N/A	V		
		V	V	V	V	V	V	V	I	N/A	V		
		V	V	V	V	V	V	V	V	I	V		
		V	V	V	V	V	V	V	V	N/A	I		
EC 1.3	La extensión de la imagen asociada no es válida.	V	V	V	V	V	I	V	V	N/A	V	<p>Muestra una señalización sobre el formulario indicando que tipo de extensión es válida para las imágenes.</p> <p>Regresa al EC 1.1</p>	Pantalla principal del Navegante(backend) /módulo Mediateca/

EC 1.4	La extensión del sonido no es válida.	V	V	V	V	V	V	V	I	N/A	V	Muestra una señalización sobre el formulario indicando que tipo de extensión es válida para los sonidos. Regresa al EC 1.1	Pantalla principal del Navegante(backend) /módulo Mediateca/
EC 1.5	Regresar al listado.											Muestra un listado con los sonidos existentes.	Pantalla principal del Navegante(backend) /módulo Mediateca/

Tabla 14: Matriz de datos caso de prueba del CU Gestionar Sonido SC Modificar sonido.

SC #3: Eliminar sonido

Escenario	Descripción	Respuesta del sistema	Flujo central
EC 1	Selecciona la opción eliminar sonido.	Muestra un mensaje de advertencia para confirmar la eliminación del sonido y permite seleccionar las opciones. <ul style="list-style-type: none"> • Aceptar • Cancelar 	Pantalla principal del Navegante(backend) /módulo Mediateca/
EC 1.1	Se selecciona la opción Aceptar.	Elimina el sonido y muestra un mensaje indicando que el sonido seleccionado ha sido eliminado correctamente.	Pantalla principal del Navegante(backend) /módulo Mediateca/Botón Aceptar
EC 1.3	Se selecciona la opción Cancelar.	Muestra el listado de los sonidos.	Pantalla principal del Navegante(backend) /módulo Mediateca/Botón Cancelar

Tabla 15: Matriz de datos caso de prueba del CU Gestionar Sonido SC Eliminar sonido.

A continuación se presenta el diseño de casos de prueba para el **CU Gestionar Diaporama**.

Descripción General

El caso de uso se inicia cuando el administrador del sistema selecciona la opción Diaporama para gestionar. El sistema muestra el listado de los diaporamas existentes y permite adicionar un nuevo diaporama, modificar los datos del diaporama seleccionado, eliminarlo y brinda la posibilidad de filtrar los diaporamas de acuerdo a un conjunto de criterios de búsqueda. El administrador selecciona una de estas opciones y finaliza el caso de uso.

Condiciones de Ejecución

Debe haberse generado el escritorio de trabajo del usuario autenticado.

Secciones a probar en el caso de uso

Nombre de la sección	Escenarios de la sección	Descripción de la funcionalidad
SC1: Adicionar diaporama.	EC 1.1: Adicionar un diaporama.	El usuario inserta todos los datos de un diaporama.
	EC1.2: Existen datos incompletos.	Muestra una señalización sobre los campos que debe llenar. Muestra una señalización sobre el formulario indicando que debe llenar los campos requeridos.
	EC1.3: Introduce código de un diaporama existente.	Muestra una señalización sobre el campo código. Muestra una señalización sobre el formulario indicando que debe llenar los campos requeridos.
SC 2: Modificar diaporama.	EC 2.1: Modificar datos de un diaporama.	El usuario modifica los datos de un diaporama.
	EC 2.2: Existen datos incompletos.	Muestra una señalización sobre los campos que debe llenar.
	EC 2.3: La extensión de la imagen asociada no es válida.	Muestra una señalización sobre el formulario indicando que tipo de extensión es válida para la imagen asociada.

	EC 2.4: La extensión del diaporama no es válida.	Muestra una señalización sobre el formulario indicando que tipo de extensión es válida para el diaporama.
	EC 2.5: Regresar al listado.	Muestra un listado con los diaporamas existentes.
SC 3: Eliminar diaporama.	EC 3.1: Eliminar diaporama.	Está seguro que desea eliminar la media seleccionada.
	EC 3.2. Se selecciona Aceptar.	Se elimina el diaporama deseado.
	EC 3.3. Se selecciona Cancelar.	Se muestra el listado de los diaporamas.

Tabla16: Secciones a probar en el CU Gestionar Diaporama.

Descripción de las variables

No	Nombre de campo	Clasificación	Valor Nulo	Descripción
1	Nombre	Campo de texto	No	Nombre de la media.
2	Descripción	Campo de texto	No	Descripción de la media.
3	Temas asociados	Campo de selección	Si	Temas asociados a la media.
4	Licencias	Lista desplegable	Si	Licencia de la media.
5	Autor	Campo de texto	No	Gestor de contenido.
6	Imagen asociada	Campo de texto	Si	Url de la imagen asociada que se va adicionar.
7	Visible	Campo booleano.	Si	Indica si la media se ve en la mediateca.

8	Media	Campo de texto	No	Url de la media que se va a adicionar.
9	Descripción tema	Campo de texto	Si	Descripción de la media para el módulo temas.
10	Código	Campo de texto	No	Código de la media.

Tabla 17: Descripción de las variables para el caso de prueba en el CU Gestionar Diaporama.

Matriz de datos

SC #1: Adicionar diaporama

ID del escenario	Escenario	Nombre	Descripción	Temas Asociados	Licencias	Autor	Imagen Asociada	Visible	Media	Descripción Tema	Código	Respuesta sistema	Flujo central
EC 1	Selecciona la opción adicionar diaporama.											Brinda la posibilidad de insertar los datos de un diaporama especificando: <ul style="list-style-type: none"> • Nombre • Descripción • Temas Asociados • Licencias • Autor • Imagen Asociada • Visible • Media • Descripción Tema • Código 	Pantalla principal del Navegante(backend) /módulo Mediateca/Interfaz Adicionar Media.
EC 1.1	Insertar datos de un diaporama	V	V	V	V	V	V	V	V	N/A	V	Valida los datos. Inserta los datos. Muestra un listado de los diaporamas.	Pantalla principal del Navegante(backend) /módulo Mediateca/Interfaz Adicionar

												<ul style="list-style-type: none"> • Asociados • Licencias • Autor • Imagen Asociada • Visible • Media • Descripción Tema • Código 	
EC 1.1	Modificar los datos de un diaporama.	V	V	V	V	V	V	V	V	N/A	V	<p>Valida los datos. Actualiza los datos. Muestra una señalización sobre el formulario indicando que ha modificado correctamente.</p>	Pantalla principal del Navegante(backend) /módulo Mediateca/ Interfaz Modificar Media/Botón Salvar/
EC 1.2	Existen datos incompletos.	I	V	V	V	V	V	V	V	N/A	V	<p>Muestra una señalización sobre los campos que debe llenar. Muestra una señalización sobre el formulario indicando que debe llenar los campos requeridos. Regresa al EC 1.1</p>	Pantalla principal del Navegante(backend) /módulo Mediateca/
		V	I	V	V	V	V	V	V	N/A	V		
		V	V	I	V	V	V	V	V	N/A	V		
		V	V	V	I	V	V	V	V	N/A	V		
		V	V	V	V	I	V	V	V	N/A	V		
		V	V	V	V	V	V	I	V	N/A	V		
		V	V	V	V	V	V	V	V	I	V		
		V	V	V	V	V	V	V	V	N/A	I		
EC 1.3	La extensión de la imagen asociada no es válida.	V	V	V	V	V	I	V	V	N/A	V	<p>Muestra una señalización sobre el formulario indicando que tipo de extensión es válida para las imágenes. Regresa al EC 1.1</p>	Pantalla principal del Navegante(backend) /módulo Mediateca/

EC 1.4	La extensión del diaporama no es válida.	V	V	V	V	V	V	V	I	N/A	V	Muestra una señalización sobre el formulario indicando que tipo de extensión es válida para los diaporamas. Regresa al EC 1.1	Pantalla principal del Navegante(backend) /módulo Mediateca/
EC 1.5	Regresar al listado											Muestra un listado con los diaporamas existentes.	Pantalla principal del Navegante(backend) /módulo Mediateca/

Tabla 19: Matriz de datos caso de prueba del CU Gestionar Diaporama SC Modificar diaporama.

SC #3: Eliminar diaporama

Escenario	Descripción	Respuesta del sistema	Flujo central
EC 1	Selecciona la opción eliminar diaporama.	Muestra un mensaje de advertencia para confirmar la eliminación del diaporama y permite seleccionar las opciones. <ul style="list-style-type: none"> • Aceptar • Cancelar 	Pantalla principal del Navegante(backend) /módulo Mediateca/
EC 1.1	Se selecciona la opción Aceptar.	Elimina el diaporama y muestra un mensaje indicando que el diaporama seleccionado ha sido eliminado correctamente.	Pantalla principal del Navegante(backend) /módulo Mediateca/Botón Aceptar
EC 1.3	Se selecciona la opción Cancelar.	Muestra el listado de los diaporamas.	Pantalla principal del Navegante(backend) /módulo Mediateca/Botón Cancelar

Tabla 20: Matriz de datos caso de prueba del CU Gestionar Diaporama SC Eliminar diaporama.

El resto de los casos de prueba realizados pueden encontrarse en el **Anexo # 1**.

3.5 RESULTADOS OBTENIDOS.

A partir de los casos de prueba se realizaron tres iteraciones, donde se utilizaron los métodos y técnicas de pruebas expuestas anteriormente, en el que se encontraron un total de 11 no conformidades, que fueron resueltas, aportando mayor calidad al Mediateca. A continuación se muestra una tabla con el resumen de las tres iteraciones realizadas.

Tipos No conformidades	1ra iteración	2da iteración	3ra iteración
Formato(A y D)			
Error técnico			
Ortografía	2		
Redacción			
Validación	2		
Opciones que no funcionan			

Error de interfaz	2		
Error de idioma			
Funcionalidad	3	2	
Excepciones			
Otros			
Recomendación			
Total	9	2	

Tabla31: Resumen de las NC encontradas.

3.6 CONCLUSIONES DEL CAPÍTULO.

En este capítulo se abordaron los métodos y técnicas existentes para la realización de pruebas. Se explicaron de modo general las pruebas de Caja Blanca y las pruebas de Caja Negra, aplicándose esta última para la validación de la solución propuesta asegurando el correcto funcionamiento del mismo.

CONCLUSIONES GENERALES

Con la culminación del presente trabajo se han dado cumplimiento a los objetivos trazados en la investigación, donde se obtuvo un módulo funcional integrado a la colección “El Navegante” y como resultado del mismo se arribaron a las siguientes conclusiones:

- ✓ Se elaboró el marco teórico de la investigación donde los sistemas similares estudiados no cumplen con las características del módulo Mediateca.
- ✓ El estudio realizado de las herramientas y tecnologías permitió la implementación de todas las funcionalidades de la versión 1.1 del módulo Mediateca, donde se generaron los artefactos fundamentales que propone la metodología RUP.
- ✓ Se validó la solución propuesta mediante pruebas de Caja Negra, basadas en los casos de prueba, que aportó una mayor calidad al producto. Todo esto permitió erradicar deficiencias encontradas y entregar un módulo perfeccionado que cumple con las necesidades requeridas por el Ministerio del Poder Popular para la Educación de Venezuela.

RECOMENDACIONES

Como resultado del proceso de investigación y realización del módulo han surgido nuevas ideas que serían recomendables tener en cuenta para un futuro perfeccionamiento del sistema:

- ✓ Estandarizar el módulo con el objetivo de integrarlo a otras plataformas educativas.
- ✓ Agregar otros tipos de recursos multimedia para enriquecer mucho más el módulo.

REFERENCIAS BIBLIOGRÁFICAS

1. I.E.S. Dr. Fleming. *Las Nuevas Tecnologías en la Educación*. [En línea] [Citado el: 3 de diciembre de 2011.] <http://nntt.informatica-fleming.com/coordina/?p=23>.
2. **Alvarez, Miguel Angel. Manual de jQuery.** [En línea] 2009. www.biblioteca-digital.net.ve/wordpress/wp-content/.../manual-jquery.pdf.
3. *La Informática educativa en el contexto actual.* **Rodríguez Lamas, Raúl.** s.l.: Edutec. Revista Electrónica de Tecnología Educativa, 2000.
4. **Núñez Rojas, Nemecio.** Concepción del Software educativo. *La Webquest, el aula virtual y el desarrollo de competencias para la investigación en los estudiantes del I Ciclo de educación – USAT, 2010.* [En línea] www.eumed.net/libros/2010a/669/.
5. **Rosario, Jimmy.** *La Tecnología de la Información y la Comunicación (TIC). Su uso como Herramienta para el Fortalecimiento y el Desarrollo de la Educación Virtual.* [En línea] <http://www.cibersociedad.net/archivo/articulo.php?art=218>.
6. **MSc. Rodríguez, Mario del Toro.** Modelos de diseño didáctico de hiperentornos de enseñanza-aprendizaje. *Resumen de la Tesis en opción al grado de Doctor en Ciencias Pedagógicas.* Ciudad de La Habana, República de Cuba : Instituto Superior Pedagógico "Enrique José Varona", 2006.
7. **Dr. C. Barreto Gelles, Iván, Dr. C. del Toro Rodríguez, Mario y MSc. Riquenes Tarragó, Aniuska.** *TECNOLOGÍA EDUCATIVA: DOS MODELOS PARA LA ACCIÓN DEL MAESTRO.* La Habana, Cuba : Ministerio de Educación Cubano, 2009.
8. **Hernández Valverde, Pedro.** *El Proceso Unificado de Rational (RUP) y su relación con las técnicas y métodos de la ingeniería de usabilidad del software.* 2005.
9. **Jacobson, Ivar, Booch, Grady y Rumbaugh, James.** *El Proceso Unificado de Desarrollo, La guía completa del Proceso Unificado, escrita por sus creadores.* Madrid : Addison Wesley, 2000.
10. IEEE Standards Association. [En línea] 15 de enero de 2011. http://standards.ieee.org/reading/ieee/std_public/description/se/1471-2000_desc.html.

11. **López, Alejandro Rivera.** *Sistema asistente para la generación de horarios de cursos.* Cholula, Puebla, México : Escuela de Ingeniería y Ciencias, Departamento de Computación, Electrónica, Física e Innovación, 16 de enero de 2008.
12. Los diferentes lenguajes de programación para la web . New Web Star. [En línea] [Citado el: 16 de Diciembre de 2010.] <http://www.newwebstar.com/ebooks/133193-los-diferentes-lenguajes-de-programaciun-para-la.html>.
13. Que son los FrameWorks. SOA Agenda. [En línea] [Citado el: 10 de Diciembre de 2010.] <http://www.soaagenda.com/journal/.../que-son-los-frameworks/>.
14. **Tavárez, David. Maestros del Web.** [En línea] 16 de Diciembre de 2011. <http://www.maestrosdelweb.com/editorial/comparacion-frameworks-javascript/>.
15. **Potencier, F. y Zaninotto, F. LibrosWeb.es .** [En línea] [Citado el: 15 de Diciembre de 2011.] http://librosweb.es/symfony_1_2/capitulo1/symfony_en_pocas_palabras.html.
16. **Ramos Pérez L, Domínguez Lovaina J, Gaviñondo Mariño X, Fresno Chávez C.** Ciudad de La Habana, Cuba : Centro Nacional de Información de Ciencias Médicas-Infomed,2008. *¿Software educativo, hipermedia o entorno educativo?* [En línea] http://bvs.sld.cu/revistas/aci/vol18_4_08/aci61008.htm..
17. **Sánchez Suárez, Jose Manuel.** [En línea] [Citado el: 10 de Diciembre de 2011.] <http://www.adictosaltrabajo.com/tutoriales/tutoriales.php?pagina=jQueryIntro>.
18. **Fernández, Luis, Pedro J. Lara, y Celia Gutiérrez.** Generación de casos de prueba a partir de especificaciones UML. [En línea] 2003. [Citado el: 22 de mayo de 2011.] <http://www.esi.uem.es/~plaraber/Investigacion/JICS2003.pdf>.
19. **Choi, E.M., y A. von Mayrhauser.** Testing object-oriented systems using extended use-cases. [En línea] 2000. [Citado el: 22 de mayo de 2011.] <http://www.informatik.uni-trier.de/~ley/db/conf/pdpta/pdpta2000.html#ChoiM00>.
20. **Pressman, Roger S.** *Ingeniería del Software. Un enfoque práctico.* España : McGraw-Hill, 2002.
21. *Software Engineering Standards.* **Wiley, John.** s.l. : IEEE Press ANSI/IEEE Standard 829-1983, 2001.

22. **Mares Amézquita, Saúl.** Gestión de No Conformidades. [En línea] 2007. [Citado el: 10 de mayo de 2011.]

<http://www.mex.opsoms.org/contenido/tuberculosis/cdtaller/presentaciones/M%C3%B3dulo%208%20>

BIBLIOGRAFÍA CONSULTADA

1. **Binder, Robert.** *Object-Oriented Software Testing. Communications of the ACM.* September, 1994.
2. **M. Sc. Orestes Coloma Rodríguez.** [En línea] [Citado el: 11 de Marzo de 2011.] <http://www.forumcyt.cu/UserFiles/forum/Textos/1101687.pdf>.
3. **Damián Pérez Valdés .** Maestros del web. Los diferentes lenguajes de programación para la web. [En línea] [Citado el: 15 de Febrero de 2011.] <http://www.maestrosdelweb.com/principiantes/los-diferentes-lenguajes-de-programacion-para-la-web/>.
4. librosweb.es. [En línea] [Citado el: 15 de Febrero de 2011.] <http://www.librosweb.es/javascript/capitulo1.html>.
5. Los diferentes lenguajes de programación para la web. New Web Star. [En línea] [Citado el: 16 de Enero de 2011.] <http://www.newwebstar.com/ebooks/133193-los-diferentes-lenguajes-de-programacion-para-la.html>.
6. maestros del web. [En línea] [Citado el: 16 de Enero de 2011.] <http://www.maestrosdelweb.com/editorial/javascript-facil-y-rapido-con-jquery/>.
7. librosweb.es. [En línea] [Citado el: Marzo de 5 de 2011.] http://www.librosweb.es/symfony_1_2/.
8. Ciberneta. Conceptos básicos de servidor Web. [En línea] [Citado el: 15 de Diciembre de 2011.] http://www.ciberneta.com/manuales/instalacion_servidor_web/1_conceptos_basicos.php.

GLOSARIO DE TÉRMINOS

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

Framework: Es una estructura de soporte definida en la cual otro proyecto de software puede ser organizado y desarrollado. Típicamente puede incluir soporte de programas, bibliotecas y un lenguaje interpretado entre otros software, para ayudar a desarrollar y unir los diferentes componentes de un proyecto. Representa una arquitectura de software que modela las relaciones generales de las entidades del dominio. Provee una estructura y una metodología de trabajo la cual extiende o utiliza las aplicaciones del dominio.

HTTP: Hipertexto Transfer Protocol / Protocolo de Transferencia de Hipertextos. Modo de comunicación para solicitar páginas Web.

HTML: Hypertext Markup Language / Lenguaje de Marcado de Hipertexto, usado para escribir documentos para servidores World Wide Web.

Multiplataforma: Es un término usado para referirse a los programas, sistemas operativos, lenguajes de programación, u otra clase de software, que puedan funcionar en diversas plataformas.

Caso de Prueba: Conjunto de condiciones o variables bajo las cuáles el analista determinará si el requisito de una aplicación es parcial o completamente satisfactorio.

IDE: Integrated Development Environment (Entorno Integrado de Desarrollo). Aplicación compuesta por un conjunto de herramientas útiles para un programador a la hora de desarrollar un software.

Metodologías: Las metodologías de desarrollo de software son utilizadas para obtener un software de alta calidad, en el tiempo planificado y que satisfaga las necesidades del cliente. En la actualidad existen muchas metodologías de desarrollo, todas creadas para solucionar problemas tanto generales como específicos de algunos tipos de proyectos de desarrollo. Estas se han dividido en metodologías ágiles y tradicionales.

Módulo: Porción funcional del software, que compone a un sistema informático. Se le denomina al conjunto de elementos fundamentales que componen el producto.

PHP: Hypertext Pre-processor, lenguaje de programación, diseñado originalmente para la creación de páginas web dinámicas. Es utilizado fundamentalmente como lenguaje interpretado del lado del servidor.

Plugins: Aplicación que añade nuevas funcionalidades a un determinado sistema ya existente.

Video: Media que brinda información audiovisual sobre un contenido. .

Aplicación web: Son aquellas aplicaciones informáticas que los usuarios pueden utilizar accediendo a un servidor web a través de internet o de una red interna mediante un navegador web.

Hipermedia: Hipertexto que no se limita a documentos escritos, sino también otros medios como imágenes, audio, video, diaporamas y animaciones.

Multimedia: Es el término que se utiliza para referirse a cualquier objeto o sistema que maneja múltiples medios de expresión (físicos o digitales) para presentar o comunicar información. De allí la expresión "multi-medios". Los medios pueden ser variados, desde textos e imágenes, hasta animaciones, sonidos y videos.

Multiplataforma: Programa o aplicación que puede utilizarse sin inconvenientes en distintas plataformas de hardware y sistemas operativos.

Recursos multimedia: Es la integración de video, imagen, animación, texto y sonido. Estos constituyen un elemento crucial en la calidad de los software educativos, lo que posibilita un gran apoyo al proceso de enseñanza – aprendizaje.

Software educativo: Es un conjunto de elementos informáticos, diseñado para ser utilizado en el proceso de enseñanza – aprendizaje. Se caracteriza por ser una aplicación altamente interactiva y por contener recursos multimedia, como videos, sonidos e imágenes; además, puede contener información sobre diferentes materias, diccionarios especializados, ejercicios y juegos instructivos que apoyan los diagnósticos y las funciones de evaluación.