

Universidad de las Ciencias Informáticas
Facultad 6



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

Título: Implementación del módulo de búsqueda para la
plataforma VideoWeb 2.0.

Autor: Edgar Alejandro Alvarez Quesada.

Tutor: Ing. Yoendry Fuentes Hernández.

La Habana, Junio del 2013

“Año 55 de la Revolución”

DECLARACIÓN DE AUTORÍA

Declaración de autoría

Declaro ser el 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 _____ del año _____.

Edgar Alejandro Alvarez Quesada

Ing. Yoendry Fuentes Hernández

Firma del autor

Firma del tutor

DATOS DE CONTACTO

Tutor:

Ing. Yoendry Fuentes Hernández.

Universidad de las Ciencias Informáticas

Email: yfuentesh@uci.cu

AGRADECIMIENTOS

A toda mi familia por brindarme siempre su apoyo y por todo el esfuerzo que han realizado para que hoy yo pueda ser un profesional.

A mi cosita linda (Amarilys) por estar siempre a mi lado, por brindarme su amor, su cariño y su comprensión.

A mi tutor Doendry por estar siempre pendiente al progreso del trabajo y ayudarme en todo lo que me hizo falta.

A mis amigos Ernesto, Danny, Lijandry, Marcial, Falcón, El Robert, El Luisma, Eliober, Osiel, por haberme brindado su amistad y en general a todos aquellas personas con las que he convivido e interactuado durante estos cinco años.

DEDICATORIA

A las tres personas más importantes en mi vida, a mis padres Juan y Luisa y a mi abuela Reina, de no ser por ustedes nunca hubiera logrado cumplir este sueño. Gracias por ser mis ejemplos a seguir como personas, por su amor y dedicación incondicional.

A toda mi familia y a todas aquellas personas que han contribuido con su granito de arena en la construcción de este sueño convertido en realidad.

RESUMEN

Con el surgimiento de internet los materiales audiovisuales han aumentado considerablemente. Varias empresas como las televisoras, productoras de cine y videotecas, utilizan los archivos multimedia (audio, video, imagen) como forma fundamental de difundir información. Todas estas empresas requieren de mecanismos de búsqueda que faciliten la recuperación de estos materiales de forma rápida y precisa.

Este trabajo constituye la realización de una versión web del módulo de búsqueda y recuperación de materiales audiovisuales para la plataforma VideoWeb 2.0, cumpliendo con los requisitos funcionales y garantizando un fácil despliegue del módulo. En la elaboración se ha hecho uso de tecnologías completamente libres, cumpliendo con las políticas de migración del país hacia software libre. Además de que constituye una alternativa factible y aplicable en cualquier entidad que necesite la búsqueda y recuperación de materiales audiovisuales.

Palabras clave

Audiovisuales, búsqueda, materiales, recuperación, web.

ÍNDICE DE CONTENIDO

Índice

Introducción.....	1
Capítulo 1: Introducción al estudio de la recuperación de materiales audiovisuales. Fundamentación teórica.....	6
Introducción.....	6
1.1 Conceptos asociados al problema.....	6
1.1.1 Multimedia.....	6
1.1.2 Búsqueda y recuperación.....	7
1.1.3 Tesouro.....	7
1.2 Procesos básicos involucrados en la búsqueda y recuperación de materiales audiovisuales.....	7
1.2.1 Descripción general.....	7
1.2.1.1 Técnicas de recuperación de información.....	8
1.2.1.2 Herramientas generales de recuperación.....	9
1.3 Descripción actual del dominio del problema.....	10
1.4 Situación problemática.....	10
1.5 Análisis de otras soluciones existentes.....	11
1.5.1 Hardata Hdx Video.....	12
1.5.2 Videoma Archivo.....	12
1.5.3 Td tm Tarsys.....	12
1.5.4 Buscador de Drupal.....	13
1.5.5 Resumen sobre las soluciones existentes.....	13
1.6 Conclusiones Parciales.....	14
Capítulo 2: Tendencias y tecnologías actuales.....	15

ÍNDICE DE CONTENIDO

Introducción.....	15
2.1 Metodología de desarrollo de software	15
2.1.1 Proceso Unificado de Desarrollo	15
2.2 Lenguaje de Modelado: UML 2.0.....	16
2.3 Herramientas <i>CASE</i>	17
2.3.1 Visual Paradigm 8.0	17
2.4 Lenguajes de Programación.....	18
2.4.1 Lenguajes del lado del cliente	19
2.4.2 Lenguajes del lado del servidor.....	21
2.5 Entorno de Desarrollo Integrado (IDE).....	21
2.6 Sistemas Gestores de Base de Datos	22
2.7 Sistemas de Administración de Contenidos (CMS).....	23
2.8 Conclusiones parciales.....	25
Capítulo 3: Presentación del módulo de búsqueda y recuperación de materiales audiovisuales.	26
Introducción.....	26
3.1 Modelo de dominio	26
3.3 Modelo de casos de uso.....	31
3.4 Conclusiones parciales.....	39
Capítulo 4: Construcción de la solución propuesta	40
Introducción.....	40
4.1 Estilos arquitectónicos	40
4.1.1 Patrones Arquitectónicos	40
4.1.2 Patrones de diseño	42
4.1.2.1 General Responsibility Assignment Software Patterns (GRASP)	42

ÍNDICE DE CONTENIDO

4.1.2.2 Gand of Four (GoF).....	43
4.2 Modelo de análisis.....	45
4.3 Modelo de diseño	46
4.3.1 Diagramas de clases del diseño.....	46
4.4 Modelo de datos.....	48
4.5 Diagrama de despliegue.....	49
4.6 Modelo de implementación.....	51
4.7 Modelo de prueba.....	53
4.7.1 Pruebas de caja negra	53
4.8 Conclusiones parciales.....	57
Conclusiones generales	59
Recomendaciones.....	60
Trabajos citados	61
Bibliografía	65
Glosario de términos	69

ÍNDICE DE TABLAS Y FIGURAS

Índice de tablas

Tabla 1: Descripción textual del caso de uso "Buscar Material".	32
Tabla 2: Caso de prueba caso de uso "Buscar Material".	54

Índice de figuras

Figura 1: Estilo arquitectónico cliente-servidor.	19
Figura 2: Modelo de dominio del módulo.	27
Figura 3: Diagrama de casos de uso del módulo.	32
Figura 4: Patrón arquitectónico MVC.	41
Figura 5: Diagrama de clases del análisis del módulo.	45
Figura 6: Diagrama de clases del diseño del módulo.	47
Figura 7: Modelo de datos del módulo.	49
Figura 8: Modelo de despliegue de la plataforma.	50
Figura 9: Diagrama de componentes del módulo.	52

Introducción

El surgimiento de Internet¹ como medio principal de comunicación electrónica de datos, ha traído consigo una gran cantidad de información de diversos tipos. Poco a poco se ha ido convirtiendo en una de las principales fuentes de generación y transmisión de información. Esto ha provocado un aumento considerable de dicha información, este aumento ha provocado que sea difícil poder manejarla. Para ello se han creado una serie de programas que permiten localizarla y recuperarla.

Ejemplo de estos programas son los motores de búsqueda, presentes en la mayoría de los sitios web, los cuales tienen como objetivo ofrecer de manera rápida y sencilla la recuperación de la información almacenada. Entre los más conocidos se encuentran *Google*², *Yahoo*³ y *Bing*⁴.

Los materiales audiovisuales forman parte de esta información y poseen gran importancia dentro de la misma. En los últimos tiempos los materiales audiovisuales se han incrementado considerablemente debido a lo fácil que se difunden. Además facilitan una mayor comprensión e interpretación de las ideas, llegando a una mayor cantidad de personas.

Para almacenar los materiales audiovisuales necesitan ir acompañados de una descripción textual que los caracterice. Esta descripción puede realizarse mediante el uso de metadatos⁵, o puede estar relacionada con la información verbal de la misma. Estas características permiten desarrollar un sistema de recuperación basado en la descripción que facilitará la búsqueda de dichos materiales.

¹ Red informática de comunicación internacional que permite el intercambio de todo tipo de información entre sus usuarios.

² Motor de búsqueda creado por Larry Page y Sergey Brin, se encuentra en la dirección <http://www.google.com>.

³ Motor de búsqueda creado por Jerry Yang y David Filo, se encuentra en la dirección <http://www.yahoo.com>.

⁴ Motor de búsqueda creado por la empresa Microsoft, se encuentra en la dirección <http://www.bing.com>.

⁵ Un metadato no es más que un dato estructurado sobre la información, o sea, información sobre información, o de forma más simple, datos sobre datos.

INTRODUCCIÓN

Cuba, al igual que el resto del mundo, ha estado inmersa en este constante desarrollo de la información, al punto de verse en la necesidad de automatizar gran parte de los centros y entidades que posee el país. Todo este proceso innovador ha provocado la necesidad de desarrollar sistemas para gestionar y recuperar dicha información.

La Universidad de las Ciencias Informáticas (UCI), como principal centro que promueve la labor social y cultural del país sobre las Tecnologías de la Información y las Comunicaciones (TIC), también se encuentra inmiscuida en este proceso de informatización. En el centro de Geoinformática y Señales Digitales (GEYSED) perteneciente a la facultad 6, existe un departamento que se especializa en el procesamiento y transmisión de las señales digitales. En este departamento se encontraban los proyectos: VideoWeb y el Sistema de Captura y Catalogación de Medias (SCCM). Se identificó que ambos proyectos poseían funcionalidades y objetivos en común, por lo que se decidió unirlos para desarrollar VideoWeb 2.0, lo que será un nuevo producto de software⁶.

La plataforma VideoWeb 1.0 fue desarrollada en Drupal 6 y posee un módulo para la recuperación de materiales audiovisuales. Este módulo es estable y configurable, pero los usuarios quedan inconformes puesto que presenta pocos criterios de búsqueda. Además se identificó que la comunidad de desarrollo de Drupal pronto dejará de dar soporte a la versión 6 y ya existe estable la versión 7 que optimiza y facilita el desarrollo. Los subsistemas de Catalogación y Recuperación también poseen un componente de búsqueda y aunque presentan diversos criterios de recuperación, deben alinearse con la elaboración del nuevo producto de software que integre todas estas funcionalidades.

Por todo lo antes expuesto se identificó como **problema a resolver**: ¿Cómo facilitar la localización de los materiales audiovisuales en la plataforma VideoWeb 2.0? La investigación tiene como **objeto de estudio** los procesos de recuperación de la información.

⁶ Conjunto de programas, instrucciones y reglas informáticas para ejecutar ciertas tareas en una computadora.

INTRODUCCIÓN

Para llevar a cabo la investigación se hace necesario el estudio de los procesos de recuperación de información de los materiales audiovisuales en la plataforma VideoWeb 1.0, lo cual constituye el **campo de acción**, planteando como **idea a defender** que se garantizará la búsqueda y recuperación de los materiales audiovisuales en el sistema.

Por lo que el **objetivo general** es desarrollar un módulo que permita la recuperación de materiales audiovisuales para la plataforma VideoWeb 2.0.

Para cumplir con el objetivo propuesto se definen las siguientes **tareas de investigación**:

1. Caracterización del proceso de búsqueda y recuperación en otros sistemas nacionales y extranjeros que soportan el proceso de recuperación de materiales audiovisuales.
2. Identificación y descripción de las actividades de búsqueda y recuperación de materiales audiovisuales en la plataforma VideoWeb 1.0.
3. Selección de las herramientas y tecnologías a utilizar en la implementación del módulo de búsqueda y recuperación para el sistema VideoWeb 2.0.
4. Elaboración de los artefactos y la documentación según la metodología de desarrollo seleccionada.
5. Desarrollo del módulo de búsqueda y recuperación de materiales audiovisuales para la plataforma VideoWeb 2.0.
6. Integración del módulo de búsqueda y recuperación de materiales audiovisuales con el resto de la plataforma VideoWeb 2.0.
7. Realización de pruebas al módulo de búsqueda y recuperación de materiales audiovisuales para la plataforma VideoWeb 2.0.

Para la realización de este trabajo de diploma y con el objetivo de lograr un mejor entendimiento del mismo se utilizaron los siguientes métodos científicos:

INTRODUCCIÓN

Métodos Teóricos

Analítico – sintético: Permitió la realización de un análisis definido del proceso de búsqueda y recuperación de los materiales audiovisuales en la plataforma VideoWeb 2.0. De esta forma se logró sintetizar la información adquirida durante el análisis realizado, facilitando así un mejor entendimiento acerca del funcionamiento de dicho proceso.

Histórico – lógico: Permitió la realización de un estudio abarcador enmarcado en la evolución de los sistemas de búsqueda y recuperación de materiales audiovisuales a nivel mundial, así como sus principales características. También facilitó organizar la información de forma lógica, para lograr un mejor entendimiento sobre la investigación realizada.

Modelación: Este método es utilizado para hacer modelos que brindan la posibilidad de crear abstracciones para explicar la realidad. El mismo se hace visible en el trabajo, cuando se crean los diagramas del módulo de búsqueda y recuperación de materiales audiovisuales para la plataforma VideoWeb 2.0. Ejemplos de estos son: el de Casos de Uso, Despliegue y Componentes.

Métodos Empíricos:

Observación: El objetivo de este método es obtener un nivel de realidad sobre los procesos que conforman el objeto de estudio, sin su utilización se corre el riesgo de no entender totalmente los procesos desarrollados en el mismo. El uso de este se evidencia cuando se observan los pasos que realiza un usuario para localizar los materiales audiovisuales. Mediante este se podrá lograr un mejor entendimiento de cómo se realiza el proceso de búsqueda y recuperación en la plataforma VideoWeb 1.0 y en los subsistemas de Catalogación y Recuperación del proyecto SCCM.

La investigación quedó estructurada de la siguiente manera:

Capítulo 1: Introducción al estudio de la recuperación de materiales audiovisuales. Fundamentación teórica. En este capítulo se tratan los conceptos esenciales

INTRODUCCIÓN

relacionados con la recuperación de materiales audiovisuales. También se explican aspectos relacionados con la situación problemática planteada y las herramientas existentes en el mundo que realizan la recuperación de materiales audiovisuales.

Capítulo 2: Tendencias y tecnologías actuales. En este capítulo se describen las tecnologías, los lenguajes de programación y la metodología de desarrollo que se seleccionó. Además se explica detalladamente el porqué de la selección de cada una de ellas.

Capítulo 3: Presentación del módulo de búsqueda y recuperación de materiales audiovisuales. En este capítulo se detalla el diagrama de dominio, los requisitos funcionales y no funcionales. Además se describen textualmente los casos de usos.

Capítulo 4: Construcción de la solución propuesta. En este capítulo se definen los diagrama de despliegue, de componentes, el estándar de codificación utilizado y también se le realizan pruebas de caja negra a la solución creada.

Capítulo 1: Introducción al estudio de la recuperación de materiales audiovisuales. Fundamentación teórica.

Introducción

En este capítulo se abarcan los conceptos esenciales relacionados con la búsqueda y recuperación de materiales audiovisuales. Se explica detalladamente los aspectos relacionados con la situación problemática planteada. Además se describen los tipos de búsquedas que se emplearán en el sistema, así como algunas de las herramientas existentes en el mundo que realizan la recuperación de medias.

1.1 Conceptos asociados al problema

Con la constante evolución de las TICs y el desarrollo de los medios informáticos, han surgido disímiles conceptos que son muy importantes y su dominio puede servir para lograr un completo y mejor entendimiento del problema descrito. A continuación se describen dichos conceptos.

1.1.1 Multimedia

Etimológicamente, la palabra multimedia significa “múltiples medios” y es utilizada en el contexto de las tecnologías de la información. Esta hace referencia a que existen “múltiples intermediarios entre la fuente y el destino de la información, es decir, que se utilizan diversos medios para almacenar, transmitir, mostrar o percibir la información” (1).

Las multimedias, utilizan conjunta y simultáneamente diversos medios, como imágenes, sonidos y texto, en la transmisión de una información (2). De una forma más sencilla, se le llama multimedia a cualquier combinación de texto, sonidos, imágenes o gráficos estáticos o en movimiento.

CAPÍTULO 1: INTRODUCCIÓN AL ESTUDIO DE LA RECUPERACIÓN DE MATERIALES AUDIOVISUALES. FUNDAMENTACIÓN TEÓRICA

1.1.2 Búsqueda y recuperación

Los procesos de búsquedas y recuperación son el conjunto de tareas mediante las cuales el usuario localiza y accede a la información que necesita para resolver su problema (3). La recuperación de información trata con la representación, el almacenamiento, la organización y el acceso a ítems de información (4).

La recuperación de información es “el conjunto de tareas mediante las cuales el usuario localiza y accede a los recursos de información que son pertinentes para la resolución del problema planteado (5). Por otro lado, El Diccionario Mac Millan de Tecnología de la Información considera a la recuperación de información como las “técnicas empleadas para almacenar y buscar grandes cantidades de datos y ponerlos a disposición de los usuarios” (6).

1.1.3 Tesauro

La norma ISO 2788-1986 define un tesauro como "un vocabulario controlado y dinámico, compuesto por términos que tienen entre ellos relaciones semánticas y genéricas y que se aplica a un dominio particular del conocimiento" (7). Por su parte, Georges Van Slype en los lenguajes de indización, define un tesauro como "una lista estructurada de conceptos. Destinados a representar de manera unívoca el contenido de los documentos y de las consultas dentro de un sistema documental determinado. Además de ayudar al usuario en la indización de los documentos y de las consultas" (8).

1.2 Procesos básicos involucrados en la búsqueda y recuperación de materiales audiovisuales

1.2.1 Descripción general

Con la evolución de las TIC la información digital se ha incrementado considerablemente. Cuando las computadoras fueron capaces de gestionar amplios volúmenes de textos, se hizo necesario elaborar disímiles estrategias para recuperar

dicha información. Debido a esto es que surgen nuevas técnicas que facilitan la búsqueda y recuperación de información de forma eficiente.

1.2.1.1 Técnicas de recuperación de información

Actualmente se utilizan técnicas que son necesarias a la hora de realizar la búsqueda y recuperación de información. Entre estas se encuentran (9):

- ✓ **Técnicas de Stemming o Técnicas de derivados:** La técnica de Stemming lo que pretende es eliminar las posibles confusiones semánticas que se puedan dar en la búsqueda de un concepto, para ello trunca la palabra y busca solo por la raíz.
- ✓ **Sistemas de recuperación de lógica difusa:** Esta técnica permite realizar consultas con frases de forma tal que se eliminan los caracteres extraños que pueden aparecer en cualquier documento tradicional.
- ✓ **Técnicas de ponderación de términos:** Esta técnica se utiliza para darle un orden de prioridad a la búsqueda que desea realizar. Los documentos buscados dependen del valor obtenido en la ponderación. El valor depende de los términos que existan en el documento y la frecuencia con que estos se repiten. Por lo que el documento más apropiado será el que contenga todos los términos de búsqueda y mayor valor repetido tenga.
- ✓ **Técnica de agrupación:** Modelo probabilístico que le asigna valores a los documentos agrupándolos por un orden de importancia. Estos valores son asignados según la frecuencia con que los documentos son recuperados teniendo en cuenta el criterio de búsqueda utilizado.

Una vez culminado el análisis de las técnicas de recuperación de información, se decidió utilizar solamente la técnica de derivados y lógica difusa, debido a que estas se pueden aplicar no solo en la recuperación de información sino también en la recuperación de datos. Por otra parte estas técnicas agilizarán el proceso de búsqueda y recuperación del módulo.

1.2.1.2 Herramientas generales de recuperación

Actualmente existen numerosas herramientas para la recuperación de la información en internet, entre las que se encuentran los buscadores, los metabuscadores, los buscadores selectivos y los agentes inteligentes. A continuación se abordará sobre las características de cada uno.

Los buscadores, también denominados motores de búsqueda, son sistemas basados en un *robot*⁷ o software que recorren de forma automática la red en busca de documentos para indexarlos y almacenarlos en una base de datos. Una vez almacenados los documentos pueden ser consultados por cualquier usuario que necesite dicha información. Entre los buscadores más utilizados se encuentran *Google*, *Yahoo*, *Bing* y *Altavista*⁸. Los metabuscadores son buscadores que no se limitan a recuperar la información en una única base de datos, sino que realizan la búsqueda en varias bases de datos. Ejemplo de estos son *Kartoo*⁹ y *Vivísimo*¹⁰.

Los buscadores selectivos utilizan una base de datos especializada en una materia en específico, ejemplos de estos son *Electric Library*¹¹, *Teoma*¹² y otros más. Por otra parte, los agentes inteligentes solo necesitan un criterio de búsqueda y el lugar donde van a realizar la recuperación, la cual la realiza de forma automática, ejemplo de estos son *Webferre*¹³ y *WebSeeker*¹⁴.

⁷ Máquina o ingenio electrónico programable, capaz de manipular objetos y realizar operaciones antes reservadas solo a las personas.

⁸ Motor de búsqueda creado por la empresa *Overture Service Inc*, se encuentra en la dirección <http://www.altavista.com>.

⁹ Metabuscador creado por Laurent Baleyrier, se encuentra en la dirección <http://www.kartoo.com>.

¹⁰ Metabuscador desarrollado en la Universidad Carnegie-Mellon, se encuentra en la dirección <http://www.vivisimo.com>.

¹¹ Buscador selectivo, se encuentra en la dirección <http://www.elibrary.com>

¹² Buscador selectivo, se encuentra en la dirección <http://www.teoma.com>

¹³ Agente inteligente, se encuentra en la dirección <http://www.ferretsoft.com>

¹⁴ Agente inteligente, se encuentra en la dirección <http://www.bluesquirrel.com>

1.3 Descripción actual del dominio del problema

El departamento de Señales Digitales perteneciente a la facultad 6 de la Universidad de las Ciencias Informáticas, actualmente se está dedicando al desarrollo de productos que trabajan con materiales audiovisuales. Por esta razón el proyecto Catalogación y Publicación de Medias quiere desarrollar un nuevo producto de software con un grupo de funcionalidades que garanticen la gestión, edición, catalogación y préstamo de las medias. Para el correcto funcionamiento de la plataforma se necesita de un módulo para la búsqueda y recuperación de la información de los materiales audiovisuales que se encuentran almacenados en una base de datos.

El proyecto SCCM posee un módulo que se encarga de localizar los materiales audiovisuales de ese proyecto, este permite buscar de dos formas diferentes, los materiales catalogados y los materiales que no se encuentran catalogados. Esta primera forma de búsqueda se puede realizar de tres maneras distintas, ellas son: básica, avanzada y por tesauros. Cuando se va a realizar el proceso de recuperación se puede hacer uso de los filtros que brinda cada búsqueda, de esta forma se obtiene una menor cantidad de resultados, posibilitando que se muestre la información más relevante para el usuario. La búsqueda de materiales no catalogados solo se utiliza cuando la documentalista necesita recuperar algún material para catalogarlo.

Por su parte Drupal también posee un módulo encargado de recuperar la información que se encuentra almacenada en la plataforma, pero este solo recupera aquellos contenidos que se encuentren publicados. Para publicar un material en la plataforma se necesita crear un nuevo tipo de contenido, en este caso sería un material audiovisual. Una vez que este se crea, se publica en la plataforma para que los usuarios puedan hacer uso de él.

1.4 Situación problemática

El departamento de Señales Digitales cuenta con dos subsistemas desarrollados para la web, ambos cuentan con un módulo para la búsqueda y recuperación de materiales audiovisuales. Ninguno de estos subsistemas cumple con los requisitos funcionales,

debido a que los criterios de búsqueda que poseen son pobres y obligan a tener un conocimiento avanzado en el negocio para poder recuperar los materiales audiovisuales.

El componente de búsqueda que posee el proyecto SCCM es estático, lo que implica que para realizar algún cambio en alguna de las búsquedas, se tenga que modificar el código fuente de la aplicación. Por otro lado no se puede integrar a la plataforma VideoWeb 2.0 ya que el mismo está desarrollado con Symfony, siendo este incompatible con Drupal, aun cuando ambos utilizan PHP como lenguaje de programación. Tampoco posee flexibilidad a la hora de realizar alguna búsqueda, debido a que obliga que se llenen la mayoría de los campos que posee el formulario. Por su parte el módulo dedicado a la localización de los materiales audiovisuales que posee VideoWeb 1.0 solo presenta dos tipos de búsquedas, la básica y la avanzada. Ambas búsquedas poseen pocos filtros y ninguna de las dos brinda la posibilidad de filtrar en un rango de fecha determinado, ni de ver si las palabras son o no sensibles a ortografía.

Por su parte el buscador de Drupal obliga a crear un nuevo tipo de contenido cada vez que se agregue un material audiovisual, además de tener que publicarlo para poder hacer uso del mismo. Aparte de que no posee filtros específicos para el tipo de contenido creado y los que posee son demasiado pobres cuando existe una gran cantidad de materiales públicos en la plataforma.

Esta aplicación web se está desarrollando con tecnologías completamente libres, cumpliendo con las políticas de migración de la UCI y el país hacia software libre. El mismo presenta un inconveniente y es que no posee un módulo para la recuperación de materiales audiovisuales, lo cual hace muy engorroso para los usuarios la búsqueda de estos.

1.5 Análisis de otras soluciones existentes

Actualmente el país posee la plataforma VideoWeb 1.0, esta se encarga de gestionar materiales audiovisuales pero no satisface completamente las necesidades. El

buscador que posee no presenta los suficientes filtros para asegurar una búsqueda con calidad y rapidez. En el mundo existen diversos software dedicados a realizar este tipo de operaciones pero los mismos son privativos y la adquisición de sus licencias son demasiado caras. A continuación se muestran algunos ejemplos de estas herramientas y sus características.

1.5.1 Hardata Hdx Video

Hardata Hdx Video es un software desarrollado para la plataforma Windows de 64 Bits, dedicado a la administración de materiales audiovisuales. Entre las principales características que posee están: una poderosa herramienta para la búsqueda y catalogación de información, almacenamiento de todo el contenido de forma centralizada, reproducción de archivos MPEG, DV, DivX, H.264. Soporta la importación de archivos directamente desde editores externos, dando la posibilidad de marcar y catalogar ahí mismo (10).

1.5.2 Videoma Archivo

Videoma Archivo es un software desarrollado para la plataforma Windows como solución a la gestión digital de audio, video e imagen. Entre sus principales funcionalidades se encuentran: la catalogación y publicación de materiales. Soporta el almacenamiento y la gestión distribuida por internet o a través de teléfonos móviles, además está diseñada para trabajar en red, a través de múltiples puestos, desde los que se pueden realizar consultas del material almacenado en el servidor central. Ofrece la posibilidad de mantener online un archivo tanto en alta como en baja calidad, sin perder la integridad referencial de calidades por código de tiempo (11).

1.5.3 Tdtm Tarsys

Tdtm Tarsys es un software desarrollado para la plataforma Windows como solución a la gestión de materiales audiovisuales. Es una herramienta orientada al usuario que asegura que cualquier contenido, donde quiera que esté almacenado, pueda ser consultado por los usuarios como un único archivo. Entre sus principales

funcionalidades se encuentra, la búsqueda y catalogación de los materiales audiovisuales. Brinda la posibilidad de descargar los materiales para su posible distribución y edición, además de poseer un reproductor (12).

1.5.4 Buscador de Drupal

Drupal por defecto posee un buscador, el cual está configurado para buscar solamente los usuarios y los contenidos que se encuentran publicados. Esto se debe a que solo estos módulos son los que implementan los hooks correspondientes para poder realizar búsquedas en Drupal. Este módulo de búsqueda realiza una fuerte recuperación de información, pero no se adapta a las necesidades que se necesita para recuperar materiales audiovisuales.

1.5.5 Resumen sobre las soluciones existentes

Cuba no hace uso de ninguno de estos sistemas de gestión audiovisual de debido a que son privativos, lo que implica un alto costo en la adquisición de las licencias, además de estar atado al servicio técnico que brinda la empresa desarrolladora. El buscador que posee Drupal es libre pero no se utiliza puesto a que no realiza la búsqueda y recuperación de materiales audiovisuales. A pesar de esto se realizó un análisis de las características de los mismos para tenerlos presentes a la hora de desarrollar el módulo de búsqueda y recuperación para la plataforma VideoWeb 2.0. De los sistemas analizados anteriormente se observó que realizan una fuerte recuperación de los materiales audiovisuales, utilizando para ello varios tipos de búsquedas y filtros. Videoma Archivo por ejemplo, realiza la búsqueda básica utilizando diferentes metadatos como son: título, nombre del archivo, duración, entre otros; brindando al usuario una mayor cantidad de resultados. Mientras que Tdtm Tarsys implementa la búsqueda por tesauros, brindándole la facilidad al usuario de no solo recuperar las medias que cumplan con el filtro aplicado, sino también, recuperar aquellas que tengan alguna relación con el filtro especificado por el usuario. Por otra parte, Hardata Hdx Video implementa la búsqueda avanzada, aplicando filtros de fechas y de textos con precisión.

Las búsquedas mencionadas anteriormente, conjuntamente con la búsqueda por contenido y la búsqueda combinada se incorporarán a la plataforma VideoWeb 2.0. De esta forma el sistema tendrá una mayor flexibilidad y precisión en la búsqueda y recuperación de materiales audiovisuales.

1.6 Conclusiones Parciales

En el presente capítulo se estudió la situación problemática existente, así como los principales conceptos asociados. Dicho análisis permitió tener una visión acerca de las propiedades que debe poseer el módulo de búsqueda y recuperación, conjuntamente con las ventajas y desventajas identificadas en las soluciones existentes. También se analizaron los procesos básicos relacionados con la búsqueda y recuperación de materiales audiovisuales. De las técnicas identificadas solo se seleccionaron la técnica de derivados y la de lógica difusa, debido a que agilizan el proceso de búsqueda y pueden ser utilizadas en la recuperación de datos.

Capítulo 2: Tendencias y tecnologías actuales.

Introducción

En este capítulo se describen las tecnologías, los lenguajes de programación y la metodología de desarrollo de software seleccionada. Además se realizará una descripción de las características, funcionalidades y ventajas que poseen cada uno de ellos.

2.1 Metodología de desarrollo de software

Las metodologías de desarrollo de software surgen ante la necesidad de utilizar una serie de procedimientos, técnicas, herramientas y soporte documental a la hora de desarrollar un producto de software. Estas pretenden guiar a los desarrolladores en la elaboración del software y son aplicables para cualquier tipo de proyecto. Su principal objetivo es elevar la calidad del producto final. Se caracterizan por plantear un grupo de pasos a seguir en la creación de un proyecto para que al final, el resultado sea predecible y eficiente. Entre las metodologías más utilizadas actualmente se encuentra Proceso Unificado de Desarrollo o *Rational Unified Process* (RUP).

Se ha determinado el uso de RUP, debido a que el equipo de desarrollo posee gran experiencia desarrollando software con esta metodología. Además teniendo en cuenta la envergadura que posee el software que se desea implementar, la documentación y los artefactos generados durante todo el proceso de desarrollo serían de gran ayuda para el equipo de trabajo.

2.1.1 Proceso Unificado de Desarrollo

RUP es uno de los procesos de desarrollo de software más utilizados para la creación de proyectos a largo plazo. Este, en conjunto con UML (Lenguaje Unificado de Modelado) se ha convertido en un estándar a nivel internacional debido a la gran cantidad de empresas e instituciones que lo utilizan. Presenta cuatro fases de desarrollo de software, las cuales son: **Inicio**, **Elaboración**, **Construcción** y

CAPÍTULO 2: TENDENCIAS Y TECNOLOGÍAS ACTUALES

Transición. La fase de **Inicio** es la encargada de determinar la visión del proyecto. La **Elaboración** permite determinar la arquitectura que se utilizará. La fase de **Construcción** pretende obtener la capacidad operacional del software durante su desarrollo. Mientras que la **Transición** lo que se desea es tener un *release*¹⁵ del proyecto en desarrollo (13).

Las tres características fundamentales que posee RUP son: dirigido por casos de uso, iterativo e incremental y centrado en la arquitectura. Al ser dirigido por casos de uso, da la posibilidad al usuario de pensar en términos importantes para él y no solo en funciones que debería tener la solución final. Es centrado en la arquitectura pues indica cómo debe ser construido el sistema y en qué orden. Además debe considerar la calidad, rendimiento, reutilización y capacidad de evolución por lo que debe ser flexible en todo el proceso de desarrollo. Es iterativo e incremental puesto a que divide el trabajo en partes más pequeñas o mini proyectos, de esta forma permite que el equilibrio entre casos de uso y arquitectura se vaya logrando durante cada mini proyecto en todo el proceso de desarrollo. Cada mini proyecto se puede ver como una iteración del cual se obtiene un incremento que produce un crecimiento en el producto (14).

RUP hace referencia a la forma en que se deben obtener, organizar y documentar los requerimientos funcionales, también permite documentar las decisiones y comunicar requerimientos del negocio. Indica cómo debe realizarse el modelamiento visual de la estructura, además del comportamiento que debe tener la arquitectura y los componentes.

2.2 Lenguaje de Modelado: UML 2.0

UML (Lenguaje de Modelado Unificado) es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad (15). Surge con el objetivo de comunicar ideas a otros desarrolladores y además sirve de apoyo en el proceso de

¹⁵ Nueva versión de una aplicación informática.

análisis de un problema. Permite a los desarrolladores visualizar su trabajo en esquemas o diagramas estandarizados. UML en poco tiempo se ha convertido en una notación estándar para la comunidad de investigadores de la ingeniería de software.

Entre los principales objetivos de UML se encuentran: visualizar, especificar, construir y documentar. Mediante la visualización se puede expresar de forma gráfica un sistema para que otros desarrolladores lo puedan entender. Además UML permite especificar cuáles son las características de un sistema antes de su construcción. Mediante los modelos especificados, se pueden construir los sistemas diseñados. También los propios elementos gráficos sirven como documentación del sistema desarrollado para su futura revisión (16).

2.3 Herramientas CASE

Las Herramientas *Computer Aided Software Engineering (CASE)* son un conjunto de programas y ayudas que brindan asistencia a los analistas, ingenieros de software y desarrolladores, durante todo el ciclo de vida de un software. Permiten incrementar la productividad y el control de la calidad en el proceso de elaboración.

Estas herramientas tienen como objetivo facilitar la realización de prototipos y el desarrollo conjunto de aplicaciones, simplificar el mantenimiento de los programas, mejorar y estandarizar la documentación. Además permite facilitar la reutilización de componentes de software y permitir un desarrollo y refinamiento visual de las aplicaciones utilizando gráficos.

2.3.1 Visual Paradigm 8.0

La herramienta *CASE* utilizada es *Visual Paradigm 8.0* debido a que es una herramienta profesional de diseño UML. Esta soporta el ciclo de vida completo del desarrollo de software: análisis y diseño orientados a objetos, construcción, pruebas y despliegue. Permite dibujar todo tipo de diagramas de clases, código inverso, generar código desde diagramas y generar documentación.

Esta herramienta provee el modelado de procesos de negocio, además que permite generar un mapeo de objetos relacionales para diferentes lenguajes de programación. Ofrece una serie de facilidades para generar informes que permiten documentar al proyecto.

Visual Paradigm es un producto de excelente calidad que soporta aplicaciones web, es fácil de instalar y actualizar, presenta varios idiomas, tiene un entorno de creación de diagramas para UML 2.1, disponibilidad en múltiples plataformas y de múltiples versiones para cada necesidad. Facilita la comunicación interna del equipo de desarrollo porque hace uso de un lenguaje estándar común (17).

2.4 Lenguajes de Programación

En la construcción del módulo de búsqueda y recuperación de materiales audiovisuales debe prevalecer el estilo arquitectónico cliente-servidor, debido a que el mismo se desarrolla para la web. Este estilo plantea el procesamiento cooperativo de la información por medio de un conjunto de procesadores, donde múltiples clientes solicitan requerimientos a uno o más servidores centrales. Dichos clientes no tienen que estar necesariamente distribuidos geográficamente (18). De esta forma se puede acceder a las aplicaciones, datos o cualquier otro recurso en diferentes plataformas. El estilo cliente-servidor está compuesto principalmente por los clientes que serían los consumidores de servicios y los servidores serían los proveedores de servicios y la infraestructura de comunicaciones.

Actualmente existen diferentes lenguajes para desarrollar aplicaciones web y se encuentran divididos principalmente en dos grupos, el primer grupo dedicado a la programación del lado del cliente y el segundo a la programación del lado del servidor. Los lenguajes de programación del lado del cliente se utilizan principalmente en la validación de los datos y en la estética de la interfaz de usuario. Mientras que los lenguajes de programación del lado del servidor son los encargados de comunicarse de forma segura y eficiente con el servidor, enviando los datos proporcionados por el usuario de forma dinámica y a su vez proporcionándole la respuesta del servidor.

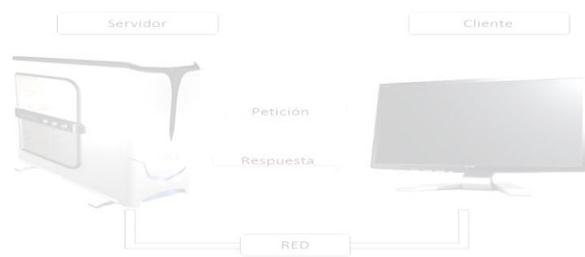


Figura 1: Estilo arquitectónico cliente-servidor.

Para el desarrollo del módulo de búsqueda y recuperación de materiales audiovisuales, se seleccionaron una serie de lenguajes de programación, tanto del lado del cliente como del servidor. Dichos lenguajes se encuentran entre los más utilizados a nivel mundial, a continuación se mencionan los mismos, conjuntamente con las características que posibilitaron su selección.

2.4.1 Lenguajes del lado del cliente

➤ **JavaScript:**

JavaScript es un lenguaje de programación interpretado del lado del cliente, lo que significa que el navegador es quién soporta la carga de procesamiento. Gracias a su compatibilidad con la mayoría de los navegadores modernos, es uno de los lenguajes de programación más utilizados. Es un lenguaje sencillo y pensado para desarrollar páginas con estética y rapidez. Permite interactividad entre el usuario y la página, además verifica los datos introducidos por el usuario antes de enviar el formulario al servidor. También es utilizado para crear contenidos dinámicos, elementos con animaciones, cambiar colores, crear menús interactivos, entre otros.

➤ **jQuery 1.7.2:**

Se utiliza la biblioteca *jQuery 1.7.2* escrita en el lenguaje *JavaScript* que le brinda a los desarrolladores un grupo de funcionalidades importantes. Dichas funcionalidades facilitan el embellecimiento y el dinamismo en las páginas, mejoran el diseño y la usabilidad y además son compatibles con la mayoría de los navegadores web.

jQuery permite la manipulación del *Document Object Model (DOM)*, contiene *API's*¹⁶ para el trabajo con *Ajax* y para manejar animaciones y efectos.

➤ **HTML 5:**

El HTML (*HyperText Markup Language*) es el lenguaje predominante en las páginas web, este lenguaje es el encargado de describir la estructura y el contenido de la página web en forma de texto. Los documentos *HTML* son hipertextos en los cuales aparecen enlaces a otros documentos, estos presentan estándares de código conocido por la mayoría de las empresas relacionadas con internet, lo que posibilita que la visualización sea prácticamente igual en todos los navegadores. *HTML 5* incorpora nuevos tipos de datos como son, *email*, *number*, *url*, *datetime*, además brinda la posibilidad de validar el contenido sin utilizar *JavaScript*.

➤ **Cascading Style Sheets (CSS):**

Las Hojas de Estilo en Cascada (*Cascading Style Sheets*), es un mecanismo simple que describe cómo se va a mostrar o a imprimir un documento en la pantalla, o incluso cómo va a ser pronunciada la información presente en ese documento a través de un dispositivo de lectura. Esta forma de descripción de estilos ofrece a los desarrolladores el control total sobre estilo y formato de sus documentos. *CSS* es utilizado para embellecer los documentos *HTML*, permitiendo separar el contenido de la presentación y además también permite controlar la estética y el formato de varias páginas web.

➤ **AJAX:**

AJAX (Asynchronous JavaScript + XML) se utiliza como método para la comunicación con la base de datos. Es la unión de varias tecnologías web que tienen como objetivo crear aplicaciones interactivas. Esta técnica se ejecuta en el lado del cliente de forma asíncrona con el servidor, posibilitando que la página web no se recargue nuevamente cuando se quiera hacer un cambio en el entorno. Esta

¹⁶ Son el conjunto de funciones y procedimientos que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción. Son usadas generalmente en las librerías.

tecnología ayuda a que las páginas web sean más eficientes, rápidas e interactivas que las que no utilizan AJAX (19).

2.4.2 Lenguajes del lado del servidor

➤ **PHP Hypertext Pre-processor (PHP 5):**

PHP 5 es un lenguaje de programación ampliamente difundido debido a que es de código abierto, se utiliza principalmente en la elaboración de sitios web grandes que tengan una fuerte administración y consultas a base de datos. A diferencia de JavaScript, PHP 5 se ejecuta en el servidor y el resultado es enviado al navegador generalmente en forma de HTML. La mayoría de los CMS (Sistemas de Administración de Contenidos) de código abierto están desarrollados sobre este lenguaje. Entre las facilidades que brinda PHP se encuentran: la integración con la mayoría de los sistemas gestores de bases de datos, la implementación de gran cantidad de funciones que agilizan su interpretación, además presenta una sintaxis flexible y no presenta tipos de datos.

2.5 Entorno de Desarrollo Integrado (IDE)

Un Entorno de Desarrollo Integrado (*IDE*) puede ser una aplicación por sí sola o puede ser parte de otra aplicación existente. Integra diferentes funciones como son: la edición, compilación, interpretación y depuración del código fuente. Además puede ser específico para un lenguaje de programación o puede incorporar más de uno.

➤ **NetBeans 7.1:**

NetBeans 7.1 es un *IDE* de desarrollo que se adapta a un gran número de lenguajes de programación, es multiplataforma, libre y contiene un grupo de características importantes como es el caso de la integración con *PHP*. Posee un editor que integra lenguajes como *JavaScript*, *HTML* y *CSS*, además presenta un sistema de escaneo para los proyectos que se crean o se cargan, reconociendo así funcionalidades y clases presentes en dichos proyectos.

Proporciona un completamiento de código eficiente, que acelera la velocidad en la elaboración de aplicaciones. Su editor de código fuente provee todas las ventajas que presenta PHP y además posee gran robustez. Ofrece líneas de comandos para la depuración posibilitando inspeccionar el código HTML sin tener que cambiar a un navegador. Permite una integración completa en términos de administración de PostgreSQL y también posee una eficiente y ágil integración con sistemas de control de versiones.

Las características antes mencionadas son las que posibilitaron que se seleccionara como IDE de desarrollo a NetBeans 7.1. Por lo que el módulo de búsqueda y recuperación para la plataforma VideoWeb 2.0 hará uso de las facilidades que brinda dicho IDE.

2.6 Sistemas Gestores de Base de Datos

Un sistema de gestión de bases de datos (SGBD; en inglés, *Database Management System: DBMS*) es un conjunto de programas que permite a los usuarios crear y mantener una base de datos. Si bien, no es imprescindible contar con un SGBD para implementar una base de datos, este software de uso general facilita el proceso de definir, construir y manipular bases de datos para diversas aplicaciones (20). Los SGBD tienen como objetivo lograr la independencia de los datos, la consistencia y la abstracción de la información. Estos sistemas permiten modificar el esquema de una base de datos, sin tener que realizar cambios en las aplicaciones que hacen uso de ella. También facilitan el ahorro de espacio al no existir tanta redundancia de datos y crean entornos de alta disponibilidad mejorando los servicios de copias de seguridad y de recuperación de fallos.

➤ PostgreSQL 9.1

PostgreSQL está considerado como uno de los gestores de bases de datos de código abierto más avanzados del mundo. Proporciona un gran número de características que normalmente sólo se encontraban en las bases de datos comerciales tales como Oracle. Además PostgreSQL básicamente es una

herramienta cliente-servidor para la gestión de bases de datos. Se considera como uno de los sistemas de bases de datos más completos destacando su soporte de transacciones, estabilidad, escalabilidad y además es multiplataforma. Aproxima los datos a un modelo objeto-relacional y es capaz de manejar complejas reglas. Ejemplos de sus avanzadas funcionalidades son consultas SQL declarativas, optimización de consultas y herencia. Este incluye la mayoría de los tipos de datos de SQL92 y SQL99, así mismo, soporta el almacenamiento de grandes objetos binarios, como imágenes, sonidos y vídeos. Posee interfaces de programación nativas para *C/C++*, *Java*, *.Net*, *Perl*, *PHP*, *Python*, *Ruby*, entre otros, además de una excepcional documentación. Es altamente escalable tanto en la cantidad bruta de datos que puede manejar como en el número de usuarios concurrentes que puede atender. Soporta casi todas las sintaxis SQL y puede almacenar grandes volúmenes de información. Cuenta con herramientas gráficas de administración y diseño de Bases de Datos tales como *PgAdmin* y *phpPgAdmin*, estas permiten que se haga sencilla la administración de las bases de datos (21).

Se decidió utilizar PostgreSQL principalmente por el elevado costo que poseen las licencias de *Oracle*. Además es un SGBD muy eficiente cuando se almacena una gran cantidad de información. Aunque su licencia es propiedad de la Universidad de California en Berkeley, es libre para utilizar, copiar, modificar y distribuir, sin importar para los fines que se aplique, tanto comercial como académico.

2.7 Sistemas de Administración de Contenidos (CMS)

Un Sistema de Gestión de Contenidos o CMS (*Content Management System*) es un programa para la gestión y publicación de un sitio Web dinámico, con actualizaciones periódicas, apariencia uniforme y diseño centrado en el usuario (22). Se utilizan para gestionar el contenido de una página Web. Dentro de una página Web se pueden distinguir dos partes: (23)

- **Presentación:** incluye todo lo que forma el estilo, estructura y disposición de la página como por ejemplo archivos HTML, hojas de estilos e imágenes.

- **Contenido:** todo lo que los autores publican en la página para darlo a conocer, ejemplo el texto y las fotografías.

Los CMS son aplicaciones prefabricadas altamente configurables que brindan la posibilidad de manipular contenidos de propósito general, aunque se pueden personalizar todo lo que se quiera. El principal objetivo de los CMS es proveer al desarrollador de una herramienta para la construcción de aplicaciones Web que manipulen contenidos de forma dinámica minimizando la necesidad de conocimientos técnicos en cuanto a programación se refiere. Por otra parte los CMS brindan a los programadores expertos, una plataforma altamente flexible para montar sus aplicaciones a través del desarrollo de *plugins* que se integran con el sistema, de esta forma se puede hacer uso provechoso de las funcionalidades que brinda la plataforma.

- **Drupal 7:**

Drupal es un CMS libre, bajo los términos de la Licencia Pública General (*GPL*), su diseño es utilizado fundamentalmente para construir y gestionar un gran número de aplicaciones web. Extiende sus funcionalidades mediante módulos. Drupal se caracteriza por ser un potente gestor de contenido, que se encarga de formar grandes comunidades de usuarios registrados que colaboran con un fin común. Sus características básicas son suficientes como para lanzar un sitio web completo, el *core*¹⁷ incluye: administración de usuarios, páginas, vocabulario, comentarios y sindicación. Además de que nos permite adaptarlo a nuestras necesidades instalando módulos creados para satisfacer necesidades específicas de nuestro trabajo. Drupal 7 se destaca por la calidad de su código y de las páginas generadas, el respeto de los estándares de la web y un énfasis especial en la usabilidad y consistencia de todo el sistema. Drupal 7 es un sistema dinámico: en lugar de almacenar sus contenidos en archivos estáticos en el sistema de ficheros del servidor de forma fija, el contenido textual de las páginas y otras configuraciones son almacenados en una base de datos y se editan utilizando un entorno Web (24).

¹⁷ Son los módulos provistos por Drupal al instalarse, algunos de ellos fueron contribuciones de la comunidad de Drupal que se incorporaron.

2.8 Conclusiones parciales

En este capítulo se seleccionaron las herramientas y tecnologías que se utilizaran en la elaboración del módulo de búsqueda y recuperación de materiales audiovisuales. Las mismas se escogieron teniendo en cuenta la experiencia que posee el equipo de trabajo utilizando estas, además de que se encuentran entre las más utilizadas a nivel mundial. Para modelar los artefactos que se van generando durante el proceso de desarrollo se usó UML 2.0 y para representar los mismos Visual Paradigm 8.0.

Capítulo 3: Presentación del módulo de búsqueda y recuperación de materiales audiovisuales.

Introducción

En este capítulo se realiza un estudio sobre el ambiente donde será implementado el sistema y los factores que permiten representar los conceptos u objetos del mundo real identificados en el mismo. También se obtienen y describen los requisitos que debe poseer el software, tanto funcional como no funcional. Se conforma el modelo de Casos de Uso y se realiza una descripción detallada de los mismos.

3.1 Modelo de dominio

“Un modelo del dominio es una representación visual de las clases conceptuales u objetos del mundo real en un dominio de interés” (25). El modelo de dominio permite centrarse en el comportamiento del sistema y en el flujo de datos que lo hace funcionar, además nos ayuda a obtener un modelo de objetos que es una representación del modelo conceptual del sistema. Describe las entidades que participan en el sistema, las relaciones y el flujo de datos que existe entre ellas, dichas entidades se mapean en clases que se componen de propiedades y métodos.

Cuando se desea efectuar la búsqueda y recuperación de los materiales audiovisuales que gestiona el software del proyecto SCCM, se realiza a través de los módulos de Catalogación o Solicitud y Préstamo. Estos son los encargados de hacer uso del buscador para realizar dicho proceso. El mismo brinda diferentes tipos de búsquedas, las cuales hacen que la recuperación se realice de forma rápida, pero se necesita de un alto conocimiento del negocio para lograr esto.

Luego de haber realizado un estudio sobre el entorno donde se encontrará el sistema, se decide realizar el modelo de dominio debido a que no se tienen bien definidos los procesos del negocio. Además el establecimiento de las reglas del funcionamiento del producto a implementar tampoco se encuentra bien definidas. A continuación se

CAPÍTULO 3: PRESENTACIÓN DEL MÓDULO DE BÚSQUEDA Y RECUPERACIÓN DE MATERIALES AUDIOVISUALES

muestra el modelo de dominio correspondiente al entorno donde estará el sistema, teniendo en cuenta los conceptos u objetos más importantes y las relaciones que existen entre estos.

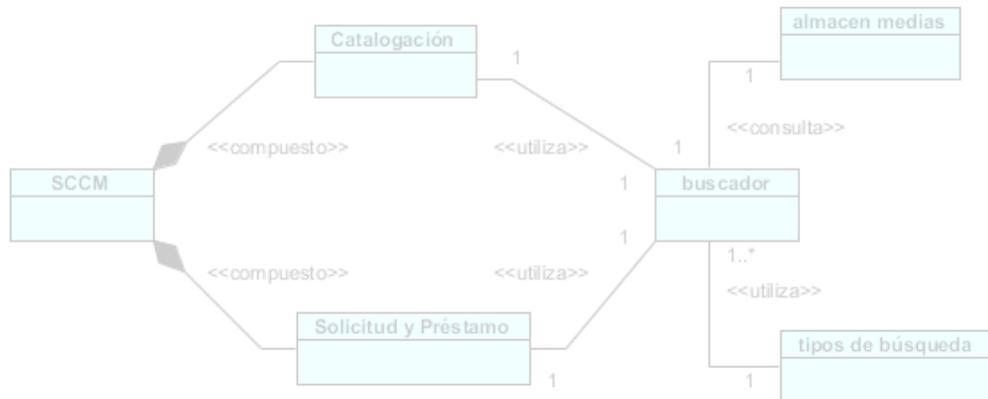


Figura 2: Modelo de dominio del módulo.

Definición de las clases del modelo de dominio:

SCCM: Sistema encargado de gestionar los materiales audiovisuales.

Catalogación y Solicitud Préstamo: Módulos que hacen uso del módulo de búsqueda y recuperación de materiales audiovisuales.

Buscador: Software encargado de realizar la búsqueda y recuperación de materiales audiovisuales.

Tipos de búsquedas: Se encuentran los diferentes tipos de búsqueda que posee el módulo de búsqueda y recuperación de materiales audiovisuales. Entre ellas se encuentran: básica, avanzada, por tesauros, por clasificación y la combina, que no es más que la unión de los filtros de las demás búsquedas. Se utiliza para brindar una única respuesta al usuario.

Bases de datos: Lugar donde se almacena la información referente a los materiales audiovisuales.

CAPÍTULO 3: PRESENTACIÓN DEL MÓDULO DE BÚSQUEDA Y RECUPERACIÓN DE MATERIALES AUDIOVISUALES

3.2 Requisitos

La principal tarea de la ingeniería de requisitos consiste en la generación de especificaciones correctas que describan con claridad, sin ambigüedades, en forma consistente y compacta, el comportamiento del sistema, de manera que le garantice minimizar los problemas relacionados con el desarrollo de software. Estas especificaciones son denominadas requisitos, los que se encuentran divididos en dos grandes grupos: los funcionales y los no funcionales. (26)

Los requisitos funcionales describen lo que el sistema debe hacer, mientras que los requisitos no funcionales son las características o restricciones que el software debe cumplir. A continuación se describen los requisitos de software que debe cumplir el módulo de búsqueda y recuperación.

Requisitos funcionales:

RF1. Realizar búsqueda básica de materiales audiovisuales.

Descripción: El sistema debe permitir la búsqueda y recuperación de los materiales audiovisuales catalogados de forma básica a partir de metadatos específicos, para ello se pueden utilizar los filtros básicos como son: el título, palabras claves, descripción, tiempo de inicio y tiempo de fin.

RF2. Realizar búsqueda avanzada de materiales audiovisuales.

Descripción: El sistema debe permitir la búsqueda y recuperación de materiales audiovisuales que se encuentran catalogados de forma avanzada. Esta búsqueda permite la utilización de filtros fecha y de texto, en el caso del texto puede buscar por todas, algunas o ninguna de las palabra que especifique y si es o no sensible a ortografía. En caso de los filtros de fecha se establecen por omisión o rango de fecha, para el rango de fecha se puede establecer una fecha de inicio y fecha de fin o alguna de las dos. Para la omisión, se puede prescindir del día, el mes o el año.

CAPÍTULO 3: PRESENTACIÓN DEL MÓDULO DE BÚSQUEDA Y RECUPERACIÓN DE MATERIALES AUDIOVISUALES

RF3. Cargar tesauros.

Descripción: El sistema debe cargar dinámicamente los tesauros existentes en el directorio, para luego buscar el o los términos en el tesoro y de esta forma poder realizar dicha búsqueda.

RF4. Buscar término en el tesoro.

Descripción: El sistema debe permitir buscar dentro del tesoro un término que pueda ser de interés para el usuario, para tenerlo en cuenta a la hora de realizar la búsqueda por tesauros.

RF5. Realizar búsqueda aplicando tesauros.

Descripción: El sistema debe permitir la búsqueda y recuperación de materiales audiovisuales que se encuentran catalogados aplicando tesauros. Mediante esta búsqueda el usuario no solo buscara por el término seleccionado sino también por las relaciones que posea dicho término, teniendo en cuenta la jerarquía de sinónimos establecida por el tesoro.

RF6. Cargar tipologías.

Descripción: El sistema debe cargar dinámicamente tipologías existentes en la base de datos para poder realizar la búsqueda por clasificación. Las tipologías no son más que los tipos de contenidos de archivos multimedia.

RF7. Realizar búsqueda por clasificación de material.

Descripción: El sistema debe permitir la búsqueda y recuperación de materiales audiovisuales que se encuentran catalogados por clasificación. Esta clasificación es el tipo de contenido del material, dígame películas, noticieros, entre otros, además de que cada contenido tiene también una serie de metadatos asociados como pueden ser: título, director, ámbito, en otros.

CAPÍTULO 3: PRESENTACIÓN DEL MÓDULO DE BÚSQUEDA Y RECUPERACIÓN DE MATERIALES AUDIOVISUALES

RF8. Realizar búsqueda combinada.

Descripción: El sistema debe permitir la búsqueda y recuperación de los materiales audiovisuales catalogados. Esta consiste en combinar todos los filtros de las búsquedas anteriormente mencionadas para brindarle al usuario una única respuesta.

RF9. Realizar búsqueda de materiales a catalogar.

Descripción: El sistema debe permitir la búsqueda y recuperación de los materiales audiovisuales que no se encuentran catalogados, aplicando una serie de filtros similares a la de la búsqueda básica y algunos particulares de esta búsqueda, como puede ser el punto de almacenamiento donde se encuentra.

RF10. Limpiar filtros de búsqueda.

Descripción: El sistema debe permitir limpiar los campos de todos los formularios de búsqueda que se encuentren activos.

Requisitos no funcionales:

➤ **RnF1.** Restricciones de diseño

Descripción: El producto de software final estará diseñado según la concepción de la arquitectura cliente-servidor en sistemas de software. Se deben respetar los estándares definidos por Drupal para la codificación del código, además de utilizar el estándar de codificación UTF-8 para la creación de tablas en la base de datos PostgreSQL.

➤ **RnF2.** Restricciones de implementación

Descripción: El lenguaje de programación que se va a utilizar para el desarrollo del módulo es PHP 5.3.

➤ **RnF3.** Restricciones de software

PC clientes

CAPÍTULO 3: PRESENTACIÓN DEL MÓDULO DE BÚSQUEDA Y RECUPERACIÓN DE MATERIALES AUDIOVISUALES

Descripción: Se recomienda un navegador como Mozilla Firefox 18, aunque se puede utilizar otro navegador web que cumpla con los estándares W3C (*World Wide Web Consortium*).

Servidores

Descripción: Se requiere sistema operativo GNU/Linux, servidor web Apache 2.2 y como Sistema Gestor de Bases de Datos PostgreSQL 9.1 o superior.

➤ **RnF4.** Restricciones de seguridad

Descripción: Para evitar el acceso a los materiales audiovisuales que puedan poseer contenidos clasificados, se implementó el *hook_permissions ()*, encargado de verificar los permisos del módulo. Por otro lado para evitar las inyecciones SQL, las consultas a la base de datos se realizan con *db_select ()*, función definida por Drupal para evitar esto.

➤ **RnF5.** Confiabilidad del sistema

- **RnF5.1:** Si se interrumpe la energía o la red en alguno de los servidores, se verán afectadas las PC clientes pues el sistema no se mostrará.
- **RnF5.2:** Al reanudarse la energía o la red, las PC clientes afectadas podrán visualizar nuevamente las funcionalidades del sistema.

3.3 Modelo de casos de uso

Los casos de uso constituyen fragmentos de funcionalidades que el sistema debe ofrecer para lograr aportar un resultado de valor a sus actores. De manera un poco más precisa, un caso de uso especifica una secuencia de acciones que el sistema puede llevar a cabo interactuando con sus actores, incluyendo alternativas dentro de la secuencia. Los diagramas de Casos de Uso del sistema constituyen una representación gráfica de los procesos y su interacción con los actores. Se utilizan para ilustrar los requisitos del sistema al mostrar cómo reacciona una respuesta a eventos que se producen en el mismo (27).

CAPÍTULO 3: PRESENTACIÓN DEL MÓDULO DE BÚSQUEDA Y RECUPERACIÓN DE MATERIALES AUDIOVISUALES

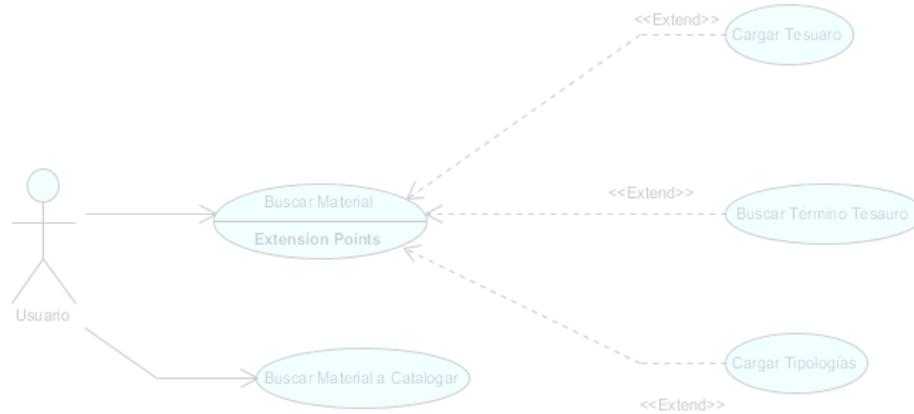


Figura 3: Diagrama de casos de uso del módulo.

La descripción de los casos de uso permite definir una serie de especificaciones que se utilizan como información de entrada en la disciplina de análisis y diseño. Esto conlleva que mientras más se detallen los elementos que componen cada caso de uso, más factible se hará el modelado de los diagramas correspondientes al análisis de la solución.

A continuación se presenta la descripción del caso de uso “Buscar Material”, dicho caso de uso es el más significativo dentro del módulo de búsqueda y recuperación. Para ver el resto de los diagramas y las descripciones de los casos de uso dirigirse al Anexo 1.

Tabla 1: Descripción textual del caso de uso "Buscar Material".

Nombre del CU	Buscar Material.
Objetivo	Mostrar materiales audiovisuales según los filtros que utilice el usuario de acuerdo al tipo de búsqueda que desea realizar.
Actores	Usuario: (Inicia)
Resumen	El caso de uso se inicia cuando se selecciona la opción de búsqueda de materiales. El sistema permite realizar cuatro tipos de búsqueda: por clasificación, avanzada, por tesauros y básica, el resultado de la búsqueda se muestra teniendo en cuenta los filtros introducidos y termina el caso de uso.
Complejidad	Alta
Prioridad	Crítico

CAPÍTULO 3: PRESENTACIÓN DEL MÓDULO DE BÚSQUEDA Y RECUPERACIÓN DE MATERIALES AUDIOVISUALES

Precondiciones	Existen archivos multimedia insertados en el sistema.	
Poscondiciones	Se obtiene el listado de materiales de acuerdo a los filtros introducidos por el usuario.	
Flujo de eventos		
Flujo básico: Buscar material.		
	Actor	Sistema
1.	Selecciona la opción “búsqueda” en la página principal.	
2.		Muestra la interfaz del buscador de materiales audiovisuales.
3.	Selecciona el tipo de búsqueda por la que desea buscar los materiales.	
4.		<p>Muestra los campos propios al tipo de búsqueda seleccionado.</p> <ul style="list-style-type: none"> - Si seleccionó la opción “Avanzada”, ver sección 1: Búsqueda avanzada. - Si seleccionó la opción “Básica”, ver sección 2: Búsqueda básica. - Si seleccionó la opción “Tesauros”, ver sección 3: Búsqueda por tesauros. - Si seleccionó la opción “Tipología”, ver sección 4: Búsqueda por clasificación.
5.	Selecciona la opción buscar.	
6.		Combina los filtros de búsqueda introducidos y muestra la lista de los materiales que cumplen con los mismos y termina el CU.
Prototipo de interfaz		

CAPÍTULO 3: PRESENTACIÓN DEL MÓDULO DE BÚSQUEDA Y RECUPERACIÓN DE MATERIALES AUDIOVISUALES

Materiales no catalogados	Básica	Avanzada	Tipología	Tesauro	Resultado
Titulo		Tiempo inicio			Tiempo fin
aceites		0			2040000
oleo		0			900000
coco		0			5220000
Flujos alternos					
5a. Evento: Limpiar campos.					
	Actor	Sistema			
5a.	Selecciona la opción Limpiar.				
		Se reinician los campos de los formularios de todos los tipos de búsqueda.			
Prototipo de interfaz					
Limpiar					
5b. Evento: Otra búsqueda.					
	Actor	Sistema			
5b.	Selecciona otro tipo de búsqueda. Ir al paso 4 del flujo normal de eventos del CU buscar material.				
Prototipo de interfaz					
Básica Avanzada Tipología Tesauro					
6a. Evento: No existen resultados.					
	Actor	Sistema			
6a.		Muestra la tabla vacía, con un mensaje indicando que no existen resultados y termina el CU.			
Prototipo de interfaz					

CAPÍTULO 3: PRESENTACIÓN DEL MÓDULO DE BÚSQUEDA Y RECUPERACIÓN DE MATERIALES AUDIOVISUALES

Materiales no catalogados Básica Avanzada Tipología Tesauro Resultado		
<div style="display: flex; justify-content: space-between; margin-bottom: 5px;"> Titulo Tempo inicio Tempo fin </div> <p style="color: gray;">La búsqueda no arrojó resultados</p>		
Sección 1: "Búsqueda avanzada".		
Flujo básico: Búsqueda avanzada.		
	Actor	Sistema
1.	<p>Introduce los datos correspondientes a los filtros de búsqueda avanzada:</p> <ul style="list-style-type: none"> ➤ Intervalo de fecha <ul style="list-style-type: none"> • Fecha inicial (Campo opcional) • Fecha final (Campo opcional) ➤ Máscara de fecha <ul style="list-style-type: none"> • Fecha (Campo opcional) • Atributo de la fecha a omitir Campo (opcional) ➤ Basada en textos <ul style="list-style-type: none"> • Texto (Campo opcional) • Ámbito (Campo opcional) • Respetar ortografía (Campo opcional) <p>Ir al paso 5 del flujo normal de los eventos del CU buscar material.</p>	
Prototipo de interfaz		

CAPÍTULO 3: PRESENTACIÓN DEL MÓDULO DE BÚSQUEDA Y RECUPERACIÓN DE MATERIALES AUDIOVISUALES

Materiales no catalogados Básica **Avanzada** Tipología Tesoro Resultado

Intervalo de fecha

Fecha inicial:

Fecha final:

Máscara de fecha

Fecha:

Omitir día:

Omitir mes:

Omitir año:

Basada en textos

Texto:

Ámbito: Seleccione:

Insensible a ortografía:

Sección 2: “Búsqueda básica”.

Flujo básico: Búsqueda básica.

	Actor	Sistema
1.	<p>Introduce los filtros correspondientes a la búsqueda básica:</p> <ul style="list-style-type: none"> ➤ Título del material (Campo opcional). ➤ Tiempo inicio (Campo opcional). ➤ Tiempo fin (Campo opcional). ➤ Descripción (Campo opcional). ➤ Palabras claves (Campo opcional). <p>Ir al paso 5 del flujo normal de los eventos del CU buscar material.</p>	

Prototipo de interfaz

CAPÍTULO 3: PRESENTACIÓN DEL MÓDULO DE BÚSQUEDA Y RECUPERACIÓN DE MATERIALES AUDIOVISUALES

Materiales no catalogados Básica Avanzada Tipología Te		
<p>Título:</p> <p>Tiempo inicio:</p> <p>Tiempo fin:</p> <p>Descripción:</p> <p>Palabras claves:</p>		
Sección 3: "Búsqueda por tesauros".		
Flujo básico: Búsqueda por tesauros.		
	Actor	Sistema
1.	Selecciona el término y las relaciones por la que desea buscar y hace clic en Adicionar, las relaciones pueden ser: <ul style="list-style-type: none"> - Más amplios. - Más restringidos. - Preferidos. - Relacionados. - Principal. - Sinónimos. <p>Observación: Este término es el que se obtiene del CU extendido Buscar término tesoro.</p>	
2.		Se adiciona el término conjuntamente con sus relaciones a la tabla de términos a buscar mostrada en la interfaz.
3.	Ir al paso 5 del flujo normal de los eventos del CU buscar material.	
Prototipo de interfaz		

CAPÍTULO 3: PRESENTACIÓN DEL MÓDULO DE BÚSQUEDA Y RECUPERACIÓN DE MATERIALES AUDIOVISUALES

Materiales no catalogados Básica Avanzada Tipología Tesauro Resultado

Seleccione término y relaciones

Tesauro... Adicionar

Más amplios:

Más restringido:

Preferidos:

Relacionados:

Principal:

Sinónimos:

Término	Tesauro	BT	NT	PT	RT	TT	SYN	
sebo de madera	Yij	X	X			X	X	✕

Flujos alternos

1a. Evento: No desea adicionar.

	Actor	Sistema
1a.	No selecciona la opción adicionar. Ir al paso 5 del flujo normal de los eventos del CU buscar material.	

Sección 4: "Búsqueda por clasificación".

Flujo básico: Búsqueda por clasificación.

	Actor	Sistema
1.		Muestra el campo seleccionable con las tipologías existentes. Observación: Ver CU extendido Cargar tipologías.
2.	Selecciona la tipología.	
3.		Muestra los campos correspondientes a la tipología seleccionada para introducir los valores.
4.	Introduce los valores deseados. Ir al paso 5 del flujo normal de los eventos del CU buscar material.	

Prototipo de interfaz

CAPÍTULO 3: PRESENTACIÓN DEL MÓDULO DE BÚSQUEDA Y RECUPERACIÓN DE MATERIALES AUDIOVISUALES

Materiales no catalogados Básica Avanzada Tipología Tesouro		
Tipología: <input style="width: 150px;" type="text" value="noticiero"/>		
noticiero		
Locutor:		
Ámbito: Seleccione: <input style="width: 50px;" type="text"/>		
Flujos alternos		
3a. Evento: No seleccionó ninguna tipología.		
	Actor	Sistema
3a.	Ir al paso 5 del flujo normal de los eventos del CU buscar material.	
Prototipo de interfaz		
Materiales no catalogados Básica Avanzada Tipología Tesouro		
Tipología: Seleccione: <input style="width: 50px;" type="text"/>		
Relaciones	CU Incluidos	No existe.
	CU Extendidos	Cargar tipologías: Paso 1 del Flujo Básico “Búsqueda por clasificación”. Buscar término en tesouro: Paso 7 del Flujo Básico “Búsqueda por tesouros”.
Requisitos no funcionales		
Asuntos pendientes	No existen.	

3.4 Conclusiones parciales

La correcta identificación de los requerimientos funcionales y no funcionales permite que se obtenga un producto, lo más consecuente con las necesidades del cliente. Por otra parte la identificación de los casos de uso a partir de los requerimientos funcionales y la descripción de estos, permiten una mejor visión del módulo que se desea construir, el comportamiento y la secuencia de actividades que debe realizar el usuario para lograr su objetivo.

Capítulo 4: Construcción de la solución propuesta

Introducción

En este capítulo se recoge la implementación del módulo de búsqueda y recuperación de materiales audiovisuales para la plataforma VideoWeb 2.0. El cual incluye la descripción de las clases, así como también los diagramas web y se especifican los patrones de diseño utilizados. Además se hace alusión a la arquitectura del módulo, al diagrama de clases persistente y al modelo de datos. También se realizan las pruebas para validar el correcto del módulo y se define el estándar de codificación a utilizar.

4.1 Estilos arquitectónicos

Un estilo arquitectónico es "Una familia de términos de un patrón de organización estructural. Más específicamente, un estilo arquitectónico determina el vocabulario de los componentes y conectores que se pueden utilizar en casos de ese estilo, junto con un conjunto de restricciones sobre la forma en que se pueden combinar" (28).

El estilo llamada y retorno permite obtener una estructura de programa que resulta relativamente fácil de modificar y cambiar de tamaño. Es el estilo más generalizado en sistemas de gran escala. Además basa su desarrollo en módulos, disminuyendo la complejidad de desarrollo significativamente. Este estilo persigue la escalabilidad y modificabilidad. Dentro del estilo llamada y retorno se encuentran los patrones arquitectónicos: En capas, basado en componentes y Modelo Vista Controlador.

4.1.1 Patrones Arquitectónicos

Los patrones arquitectónicos expresan el esquema de organización estructural fundamental para sistemas de software. Provee un conjunto de subsistemas predefinidos, especifica sus responsabilidades e incluye reglas y pautas para la organización de las relaciones entre ellos. Son plantillas para arquitecturas de software concretas, que especifican las propiedades estructurales de una aplicación y tienen un impacto en la arquitectura de subsistemas (29). Un patrón arquitectónico se enfoca en

CAPÍTULO 4: CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA

dar solución a un problema en específico, de un atributo de calidad y abarca solo parte de la arquitectura.

En un software se considera que un estilo de arquitectura, no es más que la definición de tipos particulares de componentes, así como de la interacción entre ellos, los cuales satisfacen un conjunto de restricciones. Para modelar la arquitectura del sistema se utilizó el patrón arquitectónico Modelo-Vista-Controlador (MVC), dicho modelo forma parte del estilo arquitectónico Llamada y Retorno.

Modelo-Vista-Controlador

Modelo-Vista-Controlador es un patrón de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario y la lógica de control en tres componentes distintos. Dicho patrón se ve frecuentemente en aplicaciones web, donde la vista es la página HTML y el código que provee de datos dinámicos a la página. El modelo está compuesto por el Sistema de Gestión de Base de Datos y la lógica de negocio, mientras que el controlador es el responsable de recibir los eventos de entrada desde la vista.

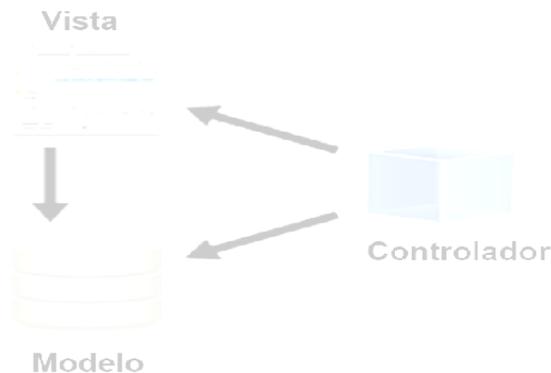


Figura 4: Patrón arquitectónico MVC.

Se decidió utilizar esta arquitectura en el módulo de búsqueda y recuperación de materiales audiovisuales, debido a que se garantiza que el modelo, la vista y el controlador se desarrollen de forma independiente. Esto permite que las modificaciones

CAPÍTULO 4: CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA

que se puedan realizar en alguna de estas partes no afecten a las demás. También permite potenciar el trabajo en equipo, eliminando el exceso de responsabilidades a un mismo desarrollador, aumentando de esta forma la rapidez y calidad con la que se desarrolla el proyecto.

4.1.2 Patrones de diseño

“Un patrón de diseño es una descripción de clases y objetos comunicándose entre sí adaptada para resolver un problema general de diseño en un contexto particular” (30). El uso de los patrones de diseño evita la búsqueda reiterada de soluciones a problemas conocidos y solucionados con anterioridad. Permite precisar un vocabulario común entre los diseñadores y además estandariza el modo con que se realiza el diseño. Dichos patrones se encuentran agrupados en dos grupos, los GRASP y los GoF.

4.1.2.1 General Responsibility Assignment Software Patterns (GRASP)

Los patrones GRASP describen los principios fundamentales de la asignación de responsabilidades a objetos, expresados en formas de patrones. El nombre se eligió para indicar la importancia de captar estos principios, con el objetivo de diseñar eficazmente el software orientado a objetos.

Alta cohesión: El patrón alta cohesión es la meta principal que ha de buscarse en todo momento, se debe tener presente en todas las decisiones de diseño. El alto nivel de cohesión es presentado por las clases que tienen responsabilidades moderadas en un área funcional y colaboran con otras para llevar a cabo las tareas, no donde dichas clases abarcan el volumen de las responsabilidades a realizar sin importar su complejidad (31).

Bajo acoplamiento: Es un patrón que tiene como principal objetivo asignar una responsabilidad para mantener el bajo acoplamiento, o sea, mantener las clases lo menos ligadas posibles. Soporta el diseño de clases más independientes, que reducen el impacto de los cambios y también más reutilizables, acrecentando la oportunidad de una mayor productividad (31).

CAPÍTULO 4: CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA

El uso de estos dos patrones se evidencia en la arquitectura seleccionada para el sistema, puesto a que ella garantiza el bajo acoplamiento y la alta cohesión.

Controlador: El patrón controlador plantea la asignación de las responsabilidades de manejar un sistema a una clase. Con el uso de este patrón se puede obtener como beneficio el incremento del potencial de los elementos que pueden ser reutilizados (31). El uso de este se evidencia en la clase Bucador_controlador.php que es la encargada de manejar el flujo de información del buscador.

Experto: Este patrón tiene como objetivo principal asignar una responsabilidad al experto en información (31). Dicho patrón ofrece como solución asignar las responsabilidades a las clases que tienen la información necesaria para cumplir con estas. Este es un principio básico que suele utilizarse en el diseño orientado a objetos. El uso de este se evidencia en las clases Tesoros.php y Perfiles.php que son las encargadas de manejar la información referente a los tesoros y los perfiles.

Creador: Este patrón tiene como objetivo principal asignar a las clases responsabilidad de crear una instancia de otra, soportando mayor claridad en el código y posibilitando el encapsulamiento y la reusabilidad (31).

El uso del mismo se evidencia en la clase Bucador_controlador.php que es la encargada de crear las instancias de las clases Bucador_modelo.php y Bucador_vista.php. Dichas clases son las encargadas de realizar las consultas a la base de datos y de crear los formularios respectivamente.

4.1.2.2 Gand of Four (GoF)

Los patrones GoF se clasifican en tres grupos principales (estructurales, creacionales y de comportamiento). Los patrones creacionales abstraen el proceso de instanciación, procuran independizar el sistema de cómo sus objetos son creados, compuestos y representados. Los patrones de comportamiento permiten conocer no solo aspectos deseables del sistema de índole estructural sino especialmente sus características

CAPÍTULO 4: CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA

dinámicas. Por último los patrones estructurales se centran en problemas relacionados con la forma de estructurar las clases.

Observador (*Observer*): Este patrón define una dependencia del tipo uno a muchos entre objetos, de manera que cuando uno de los objetos cambia su estado, el observador se encarga de notificar este cambio a todos los otros dependientes (31). El patrón observador ofrece la facilidad de no tener que estar preguntando constantemente por algún elemento, el solo se encarga de informar el cambio cuando ocurre.

El uso de este patrón se evidencia cuando algunos de los módulos que utilizan el buscador realizan alguna búsqueda, este le envía los resultados al módulo que los solicitó.

Solitario (*Singleton*): Patrón creacional a nivel de objetos. Su propósito es garantizar que una clase solo tenga una única instancia, proporcionando un punto de acceso global a la misma (31).

En general, estos objetos no encapsulan datos, lo que separa a un módulo de otro es el conjunto de funciones que contiene, por lo que debe ser pensado como una clase con una sola instancia. El uso de dicho patrón se evidencia en todos los módulos que posee Drupal y en los que se creen nuevos.

Puente (*Bridge*): Es utilizado para desacoplar una abstracción de su implementación, de manera que ambas puedan ser modificadas independientemente sin necesidad de alterar por ello la otra. Tanto la abstracción como su implementación deben ser extensibles por subclases (31).

El uso de este patrón se evidencia en la capa de abstracción de bases de datos de Drupal. Los módulos necesitan ser escritos de forma que sean independiente del sistema que se esté utilizando en la base de datos, proporcionando la capa de abstracción para ello. Una nueva capa de base de datos se puede escribir que forme

CAPÍTULO 4: CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA

parte de la API definida por el puente, añadiendo soporte para más sistemas de bases de datos sin la necesidad de modificar el código del módulo.

Acción (*Command*): Patrón de comportamiento a nivel de objetos. Su propósito es encapsular en un objeto la acción que satisface una petición, permitiendo ejecutar dicha operación sin necesidad de conocer el contenido de la misma (31).

Este patrón se usa para reducir el número de funciones que son necesarias para la aplicación, pasando la operación como un parámetro, junto con los argumentos. El uso de este se evidencia cuando se definen solamente los *hooks* que utiliza el buscador y no todos los que implementa Drupal.

4.2 Modelo de análisis

El modelo de análisis es una aproximación al modelo del diseño. Según Pressman el análisis tiene como misión el estudio de los requisitos, así como el estructurarlos y darle el refinamiento pertinente a los mismos, con el objetivo de conseguir una comprensión más precisa de ellos (32).

En este modelo no se tiene en cuenta el lenguaje de programación que se va a utilizar en la construcción de la aplicación y su objetivo es comprender los requisitos del software, no precisar cómo se implementará la solución. A continuación se muestra el diagrama de clases del análisis del módulo de búsqueda y recuperación de materiales audiovisuales.

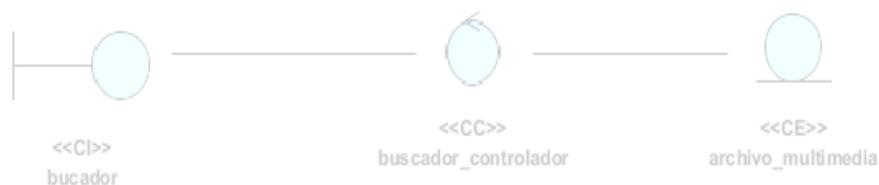


Figura 5: Diagrama de clases del análisis del módulo.

CAPÍTULO 4: CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA

4.3 Modelo de diseño

El modelo de diseño es un modelo de objetos que describe la realización física de los casos de uso del sistema, centrándose, principalmente en el impacto que tienen los requisitos funcionales y no funcionales. Este modelo sirve de abstracción entre la implementación y el sistema y por ello es utilizado como una entrada fundamental de las actividades de implementación (33).

4.3.1 Diagramas de clases del diseño

Los diagramas de clases del diseño son utilizados durante el proceso de análisis y diseño de los sistemas, donde se crea el diseño conceptual de la información que se manejará en el sistema. Un diagrama de clases presenta las clases del sistema con sus relaciones estructurales y de herencia. Dichos diagramas constituyen un elemento esencial en la concepción de cualquier aplicación, debido a que sirven de apoyo a los desarrolladores durante el desarrollo del sistema. El modelo de casos de uso aporta información para establecer las clases, objetos, atributos y operaciones (34). A continuación se muestra el diagrama de clases del diseño correspondiente al módulo de búsqueda y recuperación de materiales audiovisuales.

CAPÍTULO 4: CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA

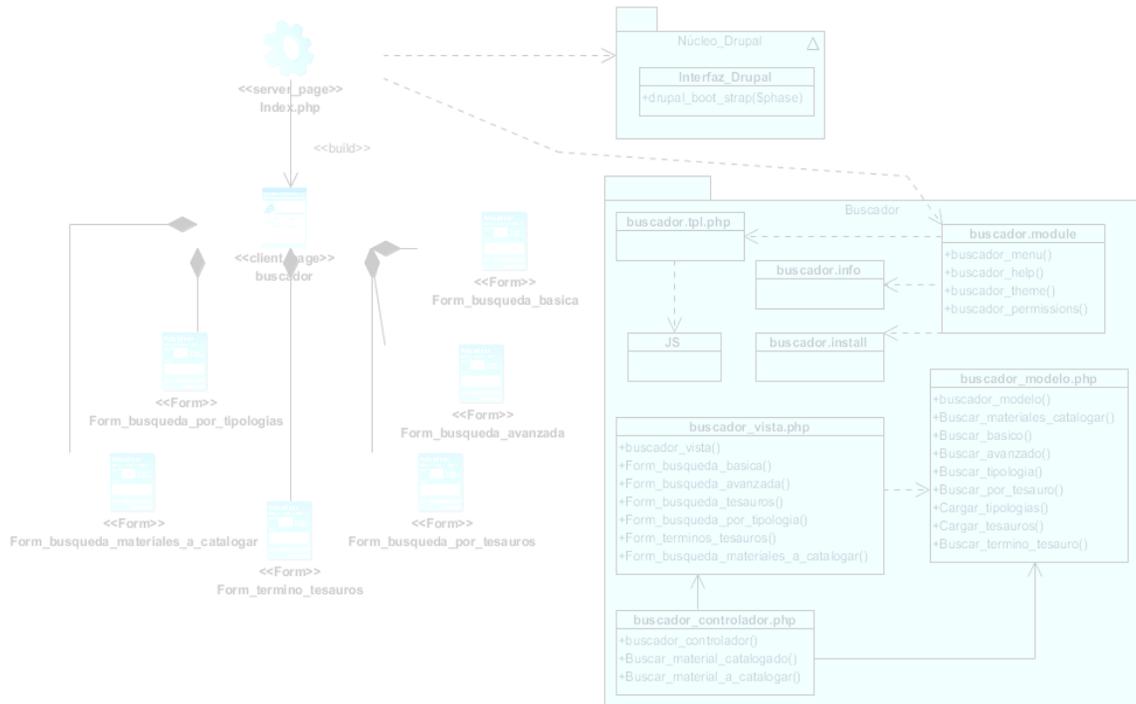


Figura 6: Diagrama de clases del diseño del módulo.

Descripción de las principales clases:

Index.php: Es la clase encargada de comunicarse con el núcleo de Drupal y el .module para construir la interfaz al usuario.

Buscador.module: Esta clase implementa una serie de funciones (*hooks*) predefinidas por Drupal para ser utilizadas durante una petición. Estos *hooks* se utilizan en la creación de los permisos que tendrá el módulo, así como en el tema que poseerá, la configuración del mismo y la ayuda que presentará dicho módulo.

Buscador.tpl.php: Es el formulario que poseerá el módulo, este se crea utilizando código HTML.

Buscador.info: Esta clase contiene los metadatos asociados al módulo, como son: título, descripción, versión de Drupal, entre otros.

CAPÍTULO 4: CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA

Buscador.install: Clase que contiene lo referente a las tablas de base datos que se necesiten crear para almacenar información referente al módulo, como puede ser: configuración o datos específicos del módulo.

JS: Clase que posee todos las funciones JavaScript del buscador.

Bucador_controladora.php: Es la clase encargada de recibir las acciones generadas por el usuario mediante su interacción con la vista y de acceder al modelo para modificarlos según la funcionalidad escogida por el usuario.

Bucador_modelo.php: Es la clase que contiene todas las consultas realizadas a la base de datos.

Bucador_vista.php: Es la clase que contiene las funciones que crean y modifican el formulario ubicado en el .tpl (*template*).

4.4 Modelo de datos

El modelo de bases de datos es la representación simple relativa, generalmente gráfica, de estructuras complejas de datos en el mundo real. La función primaria es ayudar a entender la complejidad del ambiente real y representa la estructura, características, restricciones y transformación (35).

En la siguiente figura se muestra parte del modelo de datos utilizado en el desarrollo del módulo de búsqueda y recuperación de materiales audiovisuales, perteneciente a la plataforma VideoWeb 2.0. Dicha figura está conformada por las tablas más importantes con las que interactúa el buscador.

CAPÍTULO 4: CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA

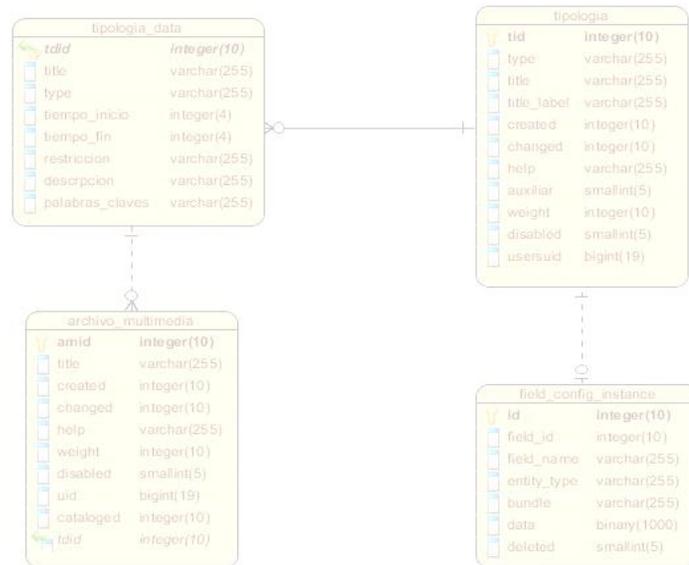


Figura 7: Modelo de datos del módulo.

En la tabla **archivo_multimedia** es donde se almacenan todos los archivos multimedia que se han subido a la plataforma, ya sean catalogados o no. La tabla **tipología** se almacenan las tipologías existentes en la plataforma, estas tipologías estarán presentes en los archivos multimedia catalogados. Por su parte la tabla **tipología_data** es la encargada de guardar la información de los archivos multimedia que se han catalogado. Esta no es más que la relación de muchos a muchos entre las tablas **archivo_multimedia** y **tipología**. Por último la tabla **field_config_instance** es la que relaciona las tipologías con los campos que poseen las mismas. Estos campos que poseen las tipologías se crean de forma dinámica y cada uno posee una tabla en la base de datos. El nombre de estas tablas se definen de la siguiente forma: **field_data_nombre_del_campo**, donde el nombre en negrita indica el nombre del campo.

4.5 Diagrama de despliegue

El Diagrama de Despliegue es un tipo de diagrama que se utiliza para modelar el hardware utilizado en la implementación de un sistema y la relación entre sus

CAPÍTULO 4: CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA

componentes. También se utiliza para capturar los elementos de configuración de procesamiento y las conexiones entre esos elementos.

Estos diagramas están compuestos por nodos, dispositivos y los conectores que los relacionan. Los nodos son objetos físicos que existen en tiempo de ejecución y representan algún tipo de recurso computacional, ya sea capacidad de memoria y/o procesamiento. Los dispositivos al igual que los nodos también se representan de la misma forma, lo que se diferencian mediante un estereotipo que los identifica (36).

Por último los conectores representan la forma mediante la cual se comunican los nodos, esta comunicación se representa mediante un estereotipo que indica el protocolo de comunicación o la red.

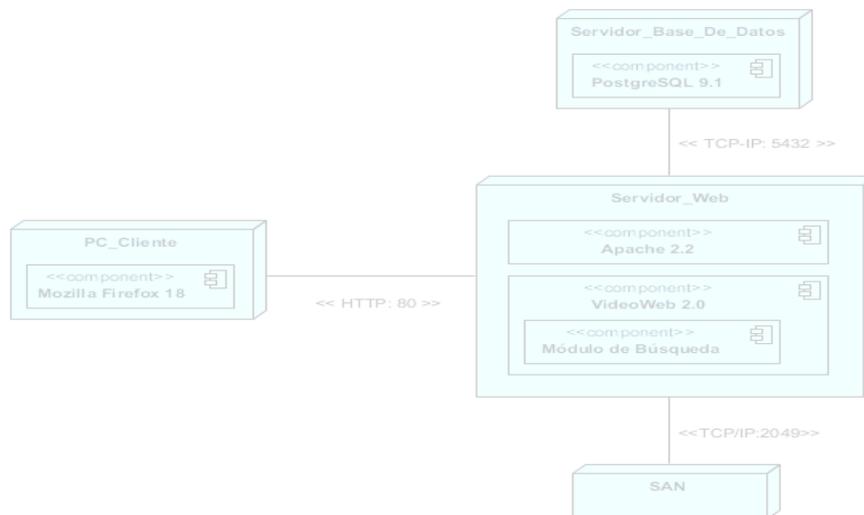


Figura 8: Modelo de despliegue de la plataforma.

Descripción de los nodos:

PC cliente: Nodo mediante el cual el cliente accederá a la aplicación utilizando un navegador web.

Servidor web: Nodo donde se encontrará instalado el servidor Web Apache, dentro del que se encuentra instalada la aplicación web perteneciente al nodo servidor web.

CAPÍTULO 4: CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA

Servidor de bases de datos: En este nodo se encontrará la base de datos del módulo y se accederá mediante la clase acceso a datos que se implementará de la aplicación web.

SAN: Servidor físico donde se almacenan los archivos multimedia.

4.6 Modelo de implementación

El modelo de implementación describe cómo se organizan los componentes de acuerdo con los mecanismos de estructuración y modularización disponibles en el entorno de implementación. Además de describirse también en el lenguaje o lenguajes de programación utilizados y cómo dependen los componentes unos de otros (37).

Diagrama de componentes:

Los diagramas de componentes describen los elementos físicos y sus realizaciones en el entorno, así como también muestran las opciones de realización. Los componentes físicos incluyen archivos, cabeceras, bibliotecas compartidas, módulos, ejecutables, o paquetes. Estos diagramas prevalecen en el campo de la arquitectura de software pero pueden ser usados para modelar y documentar cualquier arquitectura de sistema.

A continuación se presenta el diagrama de componentes:

CAPÍTULO 4: CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA

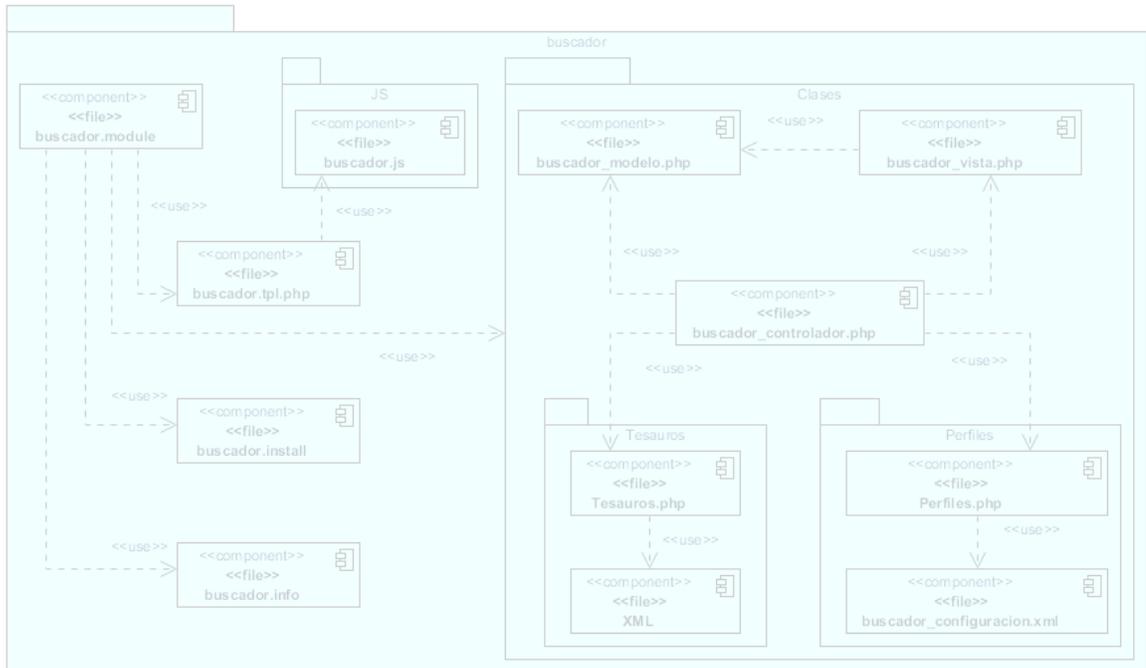


Figura 9: Diagrama de componentes del módulo.

Descripción de los paquetes y componentes:

.module: fichero que contiene funciones ganchos (*hooks*) predefinidos por Drupal para ser llamadas predeterminadas durante una petición.

.info: fichero que define los metadatos del módulo.

.install: fichero donde se crea el modelo de datos del módulo.

.tpl.php: fichero donde se define la estructura del módulo.

Clases: paquete en el que se encuentran todas las clases .php como son: la controladora, vista, modelo, tesauro y perfiles del módulo desarrollado.

JS: paquete en el que se encuentran los ficheros .js del módulo desarrollado.

CAPÍTULO 4: CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA

4.7 Modelo de prueba

En el proceso de desarrollo de software es de suma importancia verificar la calidad y el adecuado funcionamiento del software, de ahí la existencia del proceso de pruebas. La etapa de prueba es importante puesto que en ella se refleja la calidad con que ha sido llevada a cabo la construcción del sistema. En este proceso se ejecutan pruebas dirigidas a componentes del software o al sistema de software en su totalidad, teniendo como objetivo principal medir el grado en que el sistema cumple con los requisitos. O sea, comprobar que el producto se comporta como se desee. Para que las pruebas tengan éxito es necesario utilizar técnicas que guíen el proceso. Existen dos técnicas fundamentales en el proceso de pruebas que pueden realizarse en base a dos enfoques principales, estas son: las pruebas de caja blanca y las pruebas de caja negra (38).

4.7.1 Pruebas de caja negra

Las pruebas de caja negra también conocidas como pruebas funcionales o pruebas de entrada/salida, son aquellas que se llevan a cabo sobre la interfaz del software y pretenden demostrar que las funciones del software son operativas. Es decir que la entrada se acepta de forma adecuada y que se produce un resultado correcto, así como que la integridad de la información externa se mantiene. Estas pruebas se le aplicaron al módulo de búsqueda y recuperación de materiales audiovisuales para poder validar las funcionalidades que implementa el mismo.

Casos de Prueba (CP)

Los Casos de Prueba o *Test Case* son un conjunto de condiciones o variables bajo las cuáles el analista determinará si el requisito de una aplicación es parcial o completamente satisfactorio. Estos se pueden derivar de los casos de uso del sistema o del modelo de diseño, permitiendo así validar los requerimientos funcionales del sistema.

CAPÍTULO 4: CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA

Para el desarrollo de las pruebas al componente se desarrollaron cinco casos de pruebas, una por cada caso de uso del sistema. Solo se mostrará el caso de prueba realizado al caso de uso “Buscar Material” debido a que es el más significativo del sistema, los restantes casos de prueba pueden verse en el Anexo 2.

A continuación se muestra el caso de prueba realizado:

Tabla 2: Caso de prueba caso de uso “Buscar Material”.

Caso de prueba				
Nombre de la sección	Escenarios de la sección	Descripción de la funcionalidad	Resultados esperados	Resultados obtenidos
SC 1: Verificar sección buscar materiales catalogados.	EC 1.1: Obtener resultados de los materiales catalogados de forma básica.	El usuario inserta valores en la pestaña de búsqueda básica que se corresponden con las medias almacenadas y se muestran las mismas como resultado de la búsqueda.	El sistema debe mostrar el listado de materiales catalogados que cumplen con los filtros aplicados.	El sistema muestra el listado de materiales catalogados que cumplen con los filtros aplicados.
	EC 1.2: Obtener resultados de los materiales catalogados de forma avanzada.	El usuario inserta valores en la pestaña de búsqueda avanzada que se corresponden con las medias almacenadas y se muestran las mismas como resultado de la búsqueda.	El sistema debe mostrar el listado de materiales catalogados que cumplen con los filtros aplicados.	El sistema muestra el listado de materiales catalogados que cumplen con los filtros aplicados.
	EC 1.3: Obtener resultados de los materiales	El usuario inserta valores en la pestaña de búsqueda por	El sistema debe mostrar el listado de	El sistema muestra el listado de materiales

CAPÍTULO 4: CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA

	catalogados por tesauros.	tesauros que se corresponden con las medias almacenadas y se muestran las mismas como resultado de la búsqueda.	materiales catalogados que cumplen con los filtros aplicados.	catalogados que cumplen con los filtros aplicados.
	EC 1.4 Obtener resultados de los materiales catalogados por contenido.	El usuario inserta valores en la pestaña de búsqueda por contenido que se corresponden con las medias almacenadas y se muestran las mismas como resultado de la búsqueda.	El sistema debe mostrar el listado de materiales catalogados que cumplen con los filtros aplicados.	El sistema muestra el listado de materiales catalogados que cumplen con los filtros aplicados.
	EC 1.5: Obtener resultados de los materiales catalogados, combinando los filtros de todas las búsquedas.	El sistema combina los resultados de todas las búsquedas y muestra los resultados en el tab de resultados.	El sistema debe mostrar el listado de materiales catalogados combinando los resultados de todas las búsquedas.	El sistema muestra el listado de materiales catalogados combinando los resultados de todas las búsquedas.
	EC 1.6: No se obtienen resultados de los materiales catalogados de forma básica.	El usuario inserta valores en la pestaña de búsqueda básica que no se corresponden con las medias almacenadas y se	El sistema debe mostrar la tabla de resultados vacía indicando que no existen	El sistema muestra la tabla de resultados vacía indicando que no existen medias almacenadas que cumplen con los

CAPÍTULO 4: CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA

	muestra la tabla de resultados vacía.	medias almacenadas que cumplen con los filtros suministrados.	filtros suministrados.
EC 1.7: No se obtienen resultados de los materiales catalogados de forma avanzada.	El usuario inserta valores en la pestaña de búsqueda avanzada que no se corresponden con las medias almacenadas y se muestra la tabla de resultados vacía.	El sistema debe mostrar la tabla de resultados vacía indicando que no existen medias almacenadas que cumplen con los filtros suministrados.	El sistema muestra la tabla de resultados vacía indicando que no existen medias almacenadas que cumplen con los filtros suministrados.
EC 1.8: No se obtienen resultados de los materiales catalogados por tesauros.	El usuario inserta valores en la pestaña de búsqueda por tesauros que no se corresponden con las medias almacenadas y se muestra la tabla de resultados vacía.	El sistema debe mostrar la tabla de resultados vacía indicando que no existen medias almacenadas que cumplen con los filtros suministrados.	El sistema muestra la tabla de resultados vacía indicando que no existen medias almacenadas que cumplen con los filtros suministrados.
EC 1.9: No se obtienen resultados de los materiales	El usuario inserta valores en la pestaña de búsqueda por contenido que no se	El sistema debe mostrar la tabla de resultados	El sistema muestra la tabla de resultados vacía indicando que no

CAPÍTULO 4: CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA

catalogados por contenido.	corresponden con las medias almacenadas y se muestra la tabla de resultados vacía.	vacía indicando que no existen medias almacenadas que cumplan con los filtros suministrados.	existen medias almacenadas que cumplen con los filtros suministrados.
EC 1.10: Obtener resultados de los materiales catalogados, combinando los filtros de todas las búsquedas.	El sistema combina los resultados de todas las búsquedas y muestra los resultados en la pestaña de resultados.	El sistema debe mostrar el listado de materiales catalogados combinando los resultados de todas las búsquedas.	El sistema no muestra el listado de materiales catalogados combinando los resultados de todas las búsquedas.

Resultados de las pruebas:

Las pruebas de caja negra fueron realizadas de manera estricta a partir de los casos de prueba elaborados. En la culminación de este proceso se obtuvo como resultado que, exceptuando la búsqueda combinada de materiales catalogados, las funcionalidades del módulo cumplen con todos los requisitos que se definieron al inicio del proyecto. La no conformidad referente a la búsqueda combinada de los materiales catalogados fue resuelta luego de haberse detectado y actualmente funciona como se esperaba.

4.8 Conclusiones parciales

En este capítulo se definió la arquitectura que presenta el módulo de búsqueda y recuperación de materiales audiovisuales para la plataforma VideoWeb 2.0, conjuntamente con los patrones de diseño. También se elaboraron el diagrama de componentes el cual muestra la estructura física que presenta el módulo y el de

CAPÍTULO 4: CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA

despliegue. Además para validar el funcionamiento del módulo se elaboraron casos de pruebas, correspondientes a los casos de uso identificados anteriormente.

CONCLUSIONES GENERALES

Conclusiones generales

Una vez culminado el trabajo se puede concluir que:

- La caracterización de los procesos relacionados con la búsqueda y recuperación de materiales audiovisuales, en los proyectos VideoWeb y SCCM, permitió comprender el flujo de la información y las formas de recuperación. De esta forma se pudo implementar las funcionalidades correspondientes con la claridad necesaria.
- El estudio de las herramientas y tecnologías existentes, permitieron elaborar una solución informática que unificó las funcionalidades de los módulos identificados en los proyectos VideoWeb y SCCM.
- Se obtuvo como fruto primordial de este trabajo el módulo de búsqueda y recuperación de materiales audiovisuales para la plataforma VideoWeb 2.0, cumpliendo con las funcionalidades señaladas en el proyecto.
- El módulo fue construido con tecnologías completamente libres, cumpliendo con la política de migración hacia software libre del país.
- Al módulo se le aplicaron pruebas de caja negra para comprobar el funcionamiento del mismo, por lo que está listo para ser utilizado en el proyecto CPM.

RECOMENDACIONES

Recomendaciones

Al concluir el desarrollo de este trabajo se recomienda:

- Continuar desarrollando la investigación para perfeccionar y aumentar los criterios de búsquedas brindados a los usuarios.
- Desarrollar un software para gestionar los perfiles referentes a los módulos que utilizan el buscador.
- Integrar el módulo de búsqueda con el buscador que posee Drupal por defecto.

Trabajos citados

1. **Breis, Jesualdo Tomás Fernández.** *Introducción a la Multimedia y Conceptos Básicos*. 2004-2005. pág. 3. <http://dis.um.es/~jfernand/0405/tsm/tema1.pdf>.
2. **Española, Real Academia.** *Diccionario de la Real Academia de España*. s.l. : RAE. <http://lema.rae.es/drae>.
3. **Tramullas, Jesús.** *Búsqueda y recuperación*. 2000. <http://tramullas.com/documatica/3-1.html>.
4. **Baeza-Yates, R. y Ribeiro-Neto, B.** *Modern Information Retrieval*. [ed.] Addison Wesley. s.l. : ACM Press., 1999.
5. **Croft, W.Bruce.** *Approaches to intelligent information retrieval. Information Proccesing & Management*. 1987. págs. 249-254.
6. **Longley, D. and Shain M.** *Mac Millan Dictionary of IT*. s.l. : The MacMillan Press, 1989.
7. **Pérez, Sonia Collada.** *Sistema de Indexación y Búsqueda de Documentos Audiovisuales*. 2009.
8. **(ISO), International Organization for Standardization.** *Guidelines for the establishment and development of monolingual thesauri*. 1986. ISO 2788.
9. **María Pinto Molina.** *Recuperación de Información. Universidad de Granada, Facultad Biblioteconomía y Documentación.* http://www.mariapinto.es/e-coms/recu_infor.htm
10. **BMS, Grupo.** *Hardat HDX Video*. www.grupobms.com/Producto%2520HdxVideo%2520Movie%25202.pdf.
11. **ISID.** *Video Archivo*. http://www.isid.es/Spanish/product/producto_v00.htm.

TRABAJOS CITADOS

12. **Tedial**. *Td Tarsys*. <http://www.panoramaaudiovisual.com/es/2011/04/13/tedial-presenta-en-nab-sus-ultimos-avances-en-gestion-de-contenidos>.

13. **Mendoza, Maria A.** *Metodologías de desarrollo de software*. 2004.
http://www.informatizate.net/articulos/metodologias_de_desarrollo_de_software_07062004.html.

14. **Letelier, Patricio**. *Portal Desarrollo de Software*. 2008.
<http://pid.dsic.upc.es/C1/Material/default.aspx>.

15. **RedIRIS, La Forja de Conocimiento Libre de la Comunidad**. *Modelado UML*.
<http://forja.rediris.es/docman/view.php/282/444/uml20.pdf>.

16. **Orallo, Enrique Hernández**. *El Lenguaje Unificado de Modelado*. págs. 3-4.
ebernandez@disca.upv.es.

17. **Hernández, Yoendry Fuentes**. *Visual Paradigm. Diseño e implementación del módulo de recuperación de materiales audio-visuales para el Sistema de Captura y Catalogación de Medias*. 2012.

18. **Niella, R.** *Cliente-Servidor*. 2009.
http://ar.geocities.com/r_niella/Document/t_cap1.htm.

19. **Hernández, Yoendry Fuentes**. *Ajax. Diseño e implementación del módulo de recuperación de materiales audio-visuales para el Sistema Captura y Catalogación de Medias*. 2012.

20. **Pereyra, Beatriz**. *Cátedra de introducción a la Programación. Bases de Datos*. 2005.

21. **Oficial, PostgreSQL - Sitio**. *PostgreSQL*. 2010. <http://www.postgresql.org>.

22. **Juglar**. *Sistema de Gestión de Contenidos*. 2007.
<http://juglar103.blogspot.com/2004/09/01/sistema-de-gestion-de-contenidos>.

TRABAJOS CITADOS

23. **García, X. C.** *Introducción a los Sistemas de Gestión de Contenidos de código abierto*. 2004. <http://mosaic.uoc.edu/articulos/cms1204.html>.
24. **Desarrolladores, C. D.** *Sobre Drupal*. 2006. http://www.rhinocerus.net/files/About_Drupal.pdf
25. **Modelo del dominio, UML y Patrones. 2ª Edición.** Craig Larman. Prentice Hall. 2003.
26. **Falgueras, Benet Campderrich.** 2003. *Ingeniería de Software*. s.l. : Editorial UOC, 2003.
27. **Ivar Jacobson, Grady Booch, James Rumbaugh.** 2000. *El proceso unificado de desarrollo de software*. Madrid: Pearson Educación, 2000.
28. **es.scribd.** [Online] [Cited: marzo 2013, 23.] es.scribd/doc/23161581/Estilos-Arquitectónico
29. **Camacho, Erika, Cardeso, Fabio y Núñez, Gabriel.** 2004. *Arquitecturas de software*. 2004.
30. **Prieto, Felix.** *Programación III. Sistemas tema 7. Patrones de Diseño*. Universidad de Valladolid. 2005.
31. **Gamma, E., Helm, R., Johnson, R., Vlissides, J.** "Design Patterns. Elements of Reusable Object-Oriented Software". Addison-Wesley, 1995.
32. **Molina, Jesús García.** 2005. *Análisis y Diseño del Software*. 2005.
33. **Ivar Jacobson, Grady Booch, James Rumbaugh. 2000.** *El proceso unificado de desarrollo de software*. Madrid: Pearson Education, 2000.
34. *Diagramas de Clases*. http://virtual.usalesiana.edu.bo/web/practica/archiv/clases_2.doc.
35. **Dra. María G. Rosa-Rosario.** *Modelos de bases de datos*.

TRABAJOS CITADOS

36. **Pressman Roger**, 2002. Ingeniería de software. Un enfoque práctico 5ta edición.
37. **Raquel Ochoa Ornelas**, 2003. Simulador gráfico de algoritmos de programación para computadora. Universidad de Colima.
38. **Pressman, Roger S.** 2007. Ingeniería del Software. 2007.

Bibliografía

- 1 *Introducción a la Multimedia y Conceptos Básicos* 2004-2005 <http://dis.um.es/~jfernand/0405/tsm/tema1.pdf>
- 2 *Diccionario de la Real Academia de España* RAE <http://lema.rae.es/drae>
- 3 *Búsqueda y recuperación* 2000 <http://tramullas.com/documatica/3-1.html>.
- 4 *Modern Information Retrieval* ACM Press. 1999
- 5 *Approaches to intelligent information retrieval. Information Processing & Management* 1987 249-254
- 6 *Mac Millan Dictionary of IT* The MacMillan Press 1989
- 7 *Sistema de Indexación y Búsqueda de Documentos Audiovisuales* 2009
- 8 *Guidelines for the establishment and development of monolingual thesauri* 1986 ISO 2788
- 9 *Recuperación de Información.* http://www.mariapinto.es/e-coms/recu_infor.htm
- 10 *Hardat HDX*
Video www.grupobms.com/Producto%2520HdxVideo%2520Movie%25202.pdf
- 11 *Video Archivo* http://www.isid.es/Spanish/product/producto_v00.htm
- 12 *Td Tarsys* <http://www.panoramaaudiovisual.com/es/2011/04/13/tedial-presenta-en-nab-sus-ultimos-avances-en-gestion-de-contenidos>.
- 13 *Diseño e implementación del módulo de recuperación de materiales audio-visuales para el Sistema Captura y Catalogación de Medias* 2012
- 14 *Cliente-Servidor* 2009 http://ar.geocities.com/r_niella/Document/t_cap1.htm

BIBLIOGRAFÍA

- 15 *Metodologías de desarrollo de software* 2004 http://www.informatizate.net/articulos/metodologias_de_desarrollo_de_soft_ware_07062004.html
- 16 *Portal Desarrollo de Software* 2008 <http://pid.dsic.upc.es/C1/Material/default.aspx>
- 17 *Universidad de Chile. Departamento de Ciencias de La Comunicación. Taller de UML.* Chile: s.n., 1999.
- 18 *Proceso Unificado de Desarrollo de Software* 2000
- 19 *Monografias.com. Herramientas CASE para el proceso de desarrollo de Software.* <http://www.monografias.com>.
- 20 *Modelado UML* <http://forja.rediris.es/docman/view.php/282/444/uml20.pdf>
- 21 *El Lenguaje Unificado de Modelado* 3-4 bernanandez@disca.upv.es
- 22 *¿Qué es JavaScript?* [http://www.desarrolloweb.com/articulos/25.php?manual=27,\(7/1/2013\)](http://www.desarrolloweb.com/articulos/25.php?manual=27,(7/1/2013))
- 23 *MixProgramas.com. [En línea] 2012. [Citado el: 1 de febrero de 2013.]* <http://www.mixprogramas.com/netbeans-ide-7-0-beta-2/>.
- 24 <http://www.monografias.com/trabajos91/lenguaje-html/lenguaje-html.shtml>
- 25 *Diseño e implementación del módulo de recuperación de materiales audio-visuales para el Sistema de Captura y Catalogación de Medias* 2012
- 26 *Diseño e implementación del módulo de recuperación de materiales audio-visuales para el Sistema Captura y Catalogación de Medias.* 2012
- 27 *Cátedra de introducción a la Programación. Bases de Datos* 2005
- 28 *PostgreSQL* 2010 <http://www.postgresql.org>
- 29 *Sistema de Gestión de Contenidos* 2007 <http://juglar103.blogspot.com/2004/09/01/sistema-de-gestión-de-contenidos>

BIBLIOGRAFÍA

- 30 *Introducción a los Sistemas de Gestión de Contenidos de código abierto* 2004 <http://mosaic.uoc.edu/articulos/cms1204.html>
- 31 *Sobre Drupal* 2006 http://www.rhinocerus.net/files/About_Drupal.pdf
- 32 *Modelo del dominio, UML y Patrones. 2ª Edición. Craig Larman. Prentice Hall. 2003.*
- 33 *Falgueras, Benet Campderrich. 2003. Ingeniería de Software. s.l. : Editorial UOC, 2003.*
- 34 *Ivar Jacobson, Grady Booch, James Rumbaugh. 2000. El proceso unificado de desarrollo de software. Madrid: Pearson Educación, 2000.*
- 35 *es.scribd. [Online] [Cited: marzo 2013, 23.] es.scribd /doc/23161581/Estilos-Arquitectónico*
- 36 *Camacho, Erika, Cardeso, Fabio y Núñez, Gabriel. 2004. Arquitecturas de software. 2004.*
- 37 *Prieto, Felix. Programacion III. Sistemas tema 7. Patrones de Diseño. Universidad de Valladolid. 2005.*
- 38 *Molina, Jesús García. 2005. Análisis y Diseño del Software. 2005.*
- 39 *Gamma, E., Helm, R., Johnson, R., Vlissides, J. "Design Patterns. Elements of Reusable Object-Oriented Software". Addison-Wesley, 1995.*
- 40 *Dra. María G. Rosa-Rosario. Modelos de bases de datos.*
- 41 *Pressman Roger. 2002. Ingeniería de software. Un enfoque práctico 5ta edición.*
- 42 *Ivar Jacobson, Grady Booch, James Rumbaugh. 2000. El proceso unificado de desarrollo de software. Madrid: Pearson Education, 2000.*
- 43 *Diagramas de Clases.*
http://virtual.usalesiana.edu.bo/web/practica/archiv/clases_2.doc.

BIBLIOGRAFÍA

44 *Diagramas de Despliegue UML.* <http://www.slideshare.net/.../diagramas-de-despligue-uml-1475353>

45 *Raquel Ochoa Ornelas, 2003. Simulador gráfico de algoritmos de programación para computadora. Universidad de Colima. FACULTAD DE INGENIERÍA MECÁNICA Y ELÉCTRICA.*

46 *Pressman, Roger S. 2007. Ingeniería del Software. 2007.*

Glosario de términos

Agentes Inteligentes: Herramientas que recuperan la información de forma automática, solo necesitan un criterio de búsqueda y el lugar donde van a realizar la recuperación.

Altavista: Es un buscador. Se encuentra en la dirección: <http://www.altavista.com>.

Bing: Es un buscador. Se encuentra en la dirección: <http://www.bing.com>.

Buscador: Sistema basado en el uso de un software que recorre la red automáticamente para localizar documentos, indexarlos e introducirlos en una base de datos, que puede ser consultada utilizando programas de consulta.

Buscadores selectivos: Utilizan una base de datos especializada en una materia en específico.

Electric Library: Es un buscador selectivo. Se encuentra en la dirección: <http://www.elibrary.com>.

Google: Es un buscador. Se encuentra en la dirección: <http://www.google.com>.

Internet: Red informática de comunicación internacional que permite el intercambio de todo tipo de información entre sus usuarios.

Kartoo: Es un metabuscador. Se encuentra en la dirección: <http://www.kartoo.com>.

Medias: Película, imagen o cualquier otro material audio visual que requiere de un uso especial de equipamiento para visualizarlo.

Metabuscadores: Buscadores que no se limitan a recuperar la información en una única base de datos, realizan la búsqueda en diversas bases de datos.

Robot: Máquina o ingenio electrónico programable, capaz de manipular objetos y realizar operaciones antes reservadas solo a las personas.

GLOSARIO DE TÉRMINOS

Software: Conjunto de programas, instrucciones y reglas informáticas para ejecutar ciertas tareas en una computadora.

Teoma: Es un buscador selectivo. Se encuentra en la dirección: <http://www.teoma.com>.

Vivísimo: Es un metabuscador. Se encuentra en la dirección: <http://www.vivisimo.com>.

Web: Es un vocablo inglés que significa red, telaraña o malla. El concepto se utiliza en el ámbito tecnológico para nombrar a una red informática y en general a internet.

Webferret: Es un agente inteligente. Se encuentra en la dirección: <http://www.ferretsoft.com>.

WebSeeker 5: Es un agente inteligente. Se encuentra en la dirección: <http://www.bluesquirrel.com>.

Yahoo: Es un buscador. Se encuentra en la dirección: <http://www.yahoo.com>.