

Universidad de las Ciencias Informáticas



FACULTAD 6

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

Título: Módulo para la consulta y recuperación de materiales audiovisuales.

Autor(es): Yoarys Encinosa Rodríguez

Jorge Luis Suárez Falcón

Tutor: Ing. Yoendry Fuentes Hernández



FRASE



“Lo que importa verdaderamente en la vida no son los objetivos que nos marcamos, sino los caminos que seguimos para lograrlo.”

Peter Bamm

DECLARACIÓN DE AUTORÍA

Declaramos ser autores de la presente tesis y reconocemos a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste firmamos la presente a los ____ días del mes de _____ del año _____.

Yoarys Encinosa Rodríguez

Jorge Luis Suárez Falcón

Yoendry Fuentes Hernández

DATOS DE CONTACTO

Diplomantes

Nombre y apellidos: Yoarys Encinosa Rodríguez

Sexo: F **Grupo:** 6505 **Correo electrónico:** yencinosa@estudiantes.uci.cu

Nombre y apellidos: Jorge Luis Suárez Falcón

Sexo: M **Grupo:** 6504 **Correo electrónico:** jlfalcon@estudiantes.uci.cu

Tutor

Nombre y apellidos: Yoendry Fuentes Hernández

Sexo: M **Institución:** Universidad de las Ciencias Informáticas (UCI)

Dirección de la institución: Carretera a San Antonio de los Baños, Km. 2 ½, Reparto: Torrens, Municipio: Boyeros, **Provincia:** Ciudad de La Habana

Cargo del trabajador: Recién Graduado En Adiestramiento

Título de la especialidad de graduado: Ingeniero en Ciencias Informáticas

Año de graduación: 2012 **Correo electrónico:** yfuentesh@uci.cu

Institución donde se graduó: UCI

Oponente

Nombre y apellidos: Guillermo Luzua Farías.

Sexo: M **Institución:** UCI

Cargo del trabajador: Recién Graduado En Adiestramiento

Título de la especialidad de graduado: Ingeniera en Ciencias Informáticas

Año de graduación: 2012 **Correo electrónico:** gluzua@uci.cu

Institución donde se graduó: UCI

AGRADECIMIENTOS

De Jorge Luis

Quiero agradecer antes que todo a mis padres Juana y José Luis por confiar en mí, a mis tíos y tías: Manuel, Firmito, Isidro, Sandra, Gertrudis por creer en mí, nunca los voy a defraudar, a mis abuelos que junto a mis padres han sabido llevarme por el buen camino.

Quisiera agradecer también a mi segunda familia, mis compañeros de cuarto: Yaimy, Alejandro, Luis Manuel, Marcial y Edgar y a todos aquellos que vinimos juntos desde primer año por compartir aquellos buenos momentos en el apartamento, aunque nos separemos después de esto 5 años, todos recordaremos estos momentos.

Quisiera agradecer también a todas aquellas personas que con cosas buenas y malas nos educan y nos preparan para la vida.

De manera general a todas aquellas personas que han permitido que un momento tan especial como esté se esté llevando a cabo hoy.

“Gracias a todos ellos”

AGRADECIMIENTOS

De Yoarys

Quiero agradecer primeramente a mi mamá Martica que se ha dedicado y lo ha dado todo por mí, por apoyarme y por quererme siempre y a pesar de todo “Gracias mami”. También a mi abuela María Isabel (Cusa) que siempre me ha aconsejado, y me ha dado su amor de manera incondicional.

A mis padres, y digo padres porque tengo dos, mi papá Jorge Félix que no se encuentra en estos momentos, pero espero que esté orgulloso de mí y a mi otro padre Vitico que viene conmigo desde que tengo 5 años, dándome lo mejor de él, muchas gracias por la confianza depositada en mí. Gracias además a mis tíos Ihosvany, Vicente, Eduardito, Carmen por su ayuda y apoyo todos estos años, a mis primas, a mi abuelo Albertico que lo quiero mucho también. De manera general a toda mi familia por guiarme y estar a mi lado siempre.

A mi novio Luis Manuel (Tata) que son casi 6 años juntos y nunca me ha faltado su apoyo y cariño, ha estado a mi lado dándome ánimo y fuerza para seguir adelante. A su familia también pues siempre han estado pendientes de mí y me han dado todo su apoyo.

Agradecer además a dos personas que no se encuentran físicamente en este mundo pero que siempre los llevo en mi corazón, espero que estén donde estén se sientan orgullosos de mí porque sé que ellos guiaron mi camino: mis abuelos Nimia y Argelio.

A 3 personas muy importantes para mí que jugaron un papel significativo en los 5 años de la carrera y son mis amigas Mahe, Ary y Dimelsa presentes en los buenos y malos momentos, principalmente en primer año que fue un poco difícil, a todas mis amistades Leydi, Yanela, Yanet, Aindamais (Damais), Lisbet, Yalina, Nivaldo, Arturito, Robe, Luis, Ulises, Rubén, es decir a los del viejo grupo y los del nuevo también, en fin a todos los que un momento determinado compartieron conmigo.

Al profesor Yoandri Quintana que fue nuestro primer tutor y nos representó en todo momento.

No puede faltar nuestro tutor Yoendry Fuentes que nos guió, nos preparó y nos ayudó para este momento tan importante.

De manera general a todas aquellas personas que formaron o forman parte de mi vida, a todos.

“Gracias”.

DEDICATORIA

Se la dedico a toda mi familia, muy en especial a mi mamá y a mis dos padres. A mis hermanos, que los quiero con la vida. También a mis abuelos, a mi novio que siempre me estuvo apoyando en los buenos y malos momentos y a mis tíos y primas.

Yoarys

Este trabajo está dedicado a toda mi familia, en especial a mis abuelos, mis padres y a mis tíos, ellos son mi motor impulsor, y especialmente a una persona muy importante, mi hermano José Manuel que siempre está pendiente de mí.

Jorge Luis

RESUMEN

En el actual siglo los sistemas de consulta y recuperación de materiales audiovisuales han pasado a formar parte importante de la sociedad, principalmente en aquellas personas que trabajan en entidades como televisoras, productoras de cine y videotecas. Estas aplicaciones, a partir de la demanda que existe sobre las medias (audio, video o imagen) han evolucionado con el fin de encontrar la información que el usuario necesita. Teniendo en cuenta esas necesidades se han creado diferentes software con el objetivo de solicitar y recuperar materiales audiovisuales, los cuales no satisfacen las necesidades actuales de Cuba y la universidad, dichas soluciones no cuentan con las características que se necesitan para permitir una mejor gestión de los materiales audiovisuales. El presente trabajo constituye el desarrollo del módulo web para la consulta y recuperación de materiales audiovisuales del proyecto Catalogación y publicación de medias. La propuesta para la implementación se basa en el uso de herramientas y tecnologías libres, garantizando la soberanía tecnológica que propone el país.

Palabras claves

Catalogación, consulta, materiales audiovisuales, recuperación, web, medias.

ÍNDICE

INTRODUCCIÓN.....	1
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA.....	5
1.1 Términos asociados al problema.....	5
1.1.1 Materiales audiovisuales.....	5
1.1.2 Sistemas de búsqueda o recuperación de la información.....	5
1.1.3 Visualización de materiales audiovisuales.....	5
1.1.4 Solicitud de materiales audiovisuales.....	6
1.1.5 Descarga de materiales audiovisuales.....	6
1.1.6 Conversión de materiales audiovisuales.....	6
1.2 Proceso de búsqueda, visualización, solicitud y descarga de materiales audiovisuales.....	6
1.2.1 Descripción General.....	6
1.3 Análisis de otras soluciones existentes.....	7
1.3.1 YouTube.....	7
1.3.2 Metacafe.....	8
1.3.3 Stage6.....	9
1.3.4 Módulo de consulta y recuperación de materiales audiovisuales v1.0.....	10
1.4 Metodología de desarrollo de software.....	11
1.4.1 RUP.....	11
1.5 Herramientas CASE.....	13
1.5.1 Visual Paradigm.....	13
1.6.1 Modelo Cliente-Servidor.....	15
1.7 Lenguaje de programación.....	16
1.7.1 Lenguaje del lado del cliente.....	16
1.7.2 Lenguaje del lado del servidor.....	17
1.8 Sistema de gestión de contenido (CMS).....	18
1.8.1 Drupal 7.....	18
1.9 Entorno de Desarrollo Integrado (IDE).....	19
1.9.1 Netbeans.....	19

1.10 Sistemas Gestores de Bases de Datos (SGBD).....	20
1.10.1 PostgreSQL.....	20
1.11 Conclusiones Parciales.....	21
CAPÍTULO 2: CONSTRUCCIÒN DE LA SOLUCIÒN	22
2.1 Descripción del sistema	22
2.2 Diseño de la Base de Datos.....	23
2.3 Patrones Arquitectónicos	24
2.3.1 Modelo-Vista-Controlador.....	24
2.3.2 Patrones de diseño de clases	25
2.4 Diagramas de clases del diseño.....	27
2.5 Diagrama de Implementación	30
2.6 Diagrama de Despliegue	31
2.7 Estándar de codificación	32
2.8 Conclusiones parciales.....	33
CAPÍTULO 3: PRUEBA	34
3.1 Pruebas	34
3.2 Casos de Prueba	36
3.2.1 Diseño de caso de prueba para el caso de uso Gestionar Solicitud.....	36
3.2.2 Caso de prueba para el caso de uso Atender Solicitud	37
3.2.3 Caso de prueba para el caso de uso Descargar Solicitud	38
3.3 Resultados	39
3.4 Conclusiones Parciales.....	39
CONCLUSIONES GENERALES	40
RECOMENDACIONES.....	41
TRABAJOS CITADOS.....	42
ANEXOS	46
1- Requisitos Funcionales	46
1.1. Realizar búsquedas de material audiovisual.	46
1.2. Reproducir material audiovisual.	46

1.3. Gestionar Solicitud.....	48
1.3.4. Atender Solicitud	49
1.3.5. Descargar Material	49
2-Diagrama de los casos de uso del sistema	50
3- Descripción de los casos de uso.....	50
3.1- Gestionar Solicitud	50
3.2- Atender Solicitud	55
3.3- Descargar Material.....	58
GLOSARIO	64

ÍNDICE DE FIGURAS

Figura 1: Interfaz de YouTube.....	8
Figura 2 : Interfaz de Metacafe	9
Figura 3 : Interfaz de Stage 6	10
Figura 4: Modelo de datos	23
Figura 5: Diagrama de Clases del Diseño Gestionar Solicitud.....	28
Figura 6: Diagrama de Clases del Diseño Atender Solicitud.....	29
Figura 7: Diagrama de Clases del Diseño Descargar Material	30
Figura 8: Diagrama de componentes	31
Figura 9: Diagrama de despliegue	32
Figura 10: Ejemplo de código.....	33
Figura 11: Diagrama de casos de uso del sistema.....	50

ÍNDICE DE TABLAS

Tabla 1: Caso de prueba para el caso de uso gestionar solicitud.....	37
Tabla 2: Caso de prueba para el caso de uso atender solicitud.....	38
Tabla 3: Caso de prueba para el caso de uso descargar solicitud.....	39

INTRODUCCIÓN

A lo largo de la historia el hombre ha plasmado conocimientos, sentimientos y experiencias en diversos materiales para comunicarlos a sus semejantes. Los soportes físicos en los que la información se transmite han variado, condicionados por el desarrollo tecnológico de cada época, de modo que se pueden encontrar documentos en piedra, arcilla, pergamino, papel entre otros soportes. (1) Los avances tecnológicos permitieron, con la aparición de nuevos materiales y soportes (material magnético, fotográfico y plástico), que proliferaran una gran variedad de documentos desconocidos anteriormente que ya no atienden solamente al texto impreso e ilustración sino que el mundo documental se expresa en imagen fija, imagen en movimiento, archivos de ordenador y grabaciones sonoras. Entre estos se encuentran los materiales audiovisuales.

Los materiales audiovisuales en la actualidad son parte esencial de la vida diaria, convirtiéndose en gran ayuda en eventos y sucesos de trascendencia en el mundo. La gestión de estos es una actividad que en los momentos actuales requiere la intervención de sistemas informáticos, sobre todo si se tiene en cuenta la estrategia de digitalización que llevan a cabo numerosas instituciones. Para esto Cuba cuenta con el Departamento de Señales Digitales perteneciente al Centro de Desarrollo Geoinformática y Señales Digitales (GEYSED) de la Universidad de las Ciencias Informáticas (UCI), quien se encarga de la elaboración de sistemas para la gestión de audiovisuales, entre otros productos.

Este departamento posee un proyecto para la creación de la plataforma VideoWeb y otro para el Sistema de captura y catalogación de medias (SCCM), los cuales fueron integrados en un solo equipo de trabajo a finales del curso académico 2011-2012. Dicha unión se realizó pues se identificó que poseían líneas temáticas con similares propósitos. La integración de ambos sistemas trajo consigo algunos problemas arquitectónicos, uno de ellos fue que no estaban implementados en la misma tecnología, la plataforma VideoWeb usaba Drupal 6 y los módulos de SCCM estaban desarrollados en Symfony. Para esta integración, la comunicación entre ambas aplicaciones es un factor importante, sin embargo esto provocó un bajo rendimiento en la solución integral.

El propósito actual del proyecto es desarrollar la plataforma VideoWeb 2.0, que integra funcionalidades de los sistemas antes mencionados y otras. Para lograr esta integración se propone hacer uso de una misma

tecnología, buscando mejorar la comunicación entre sus módulos.

Un flujo indispensable a incluir en la nueva versión es la búsqueda, solicitud y descarga de materiales audiovisuales, flujo que en VideoWeb 1.0 no está completo. En el proyecto SCCM ya existía la concepción de un módulo con prestaciones similares a las antes mencionadas, pero la misma no cuenta con todas las funcionalidades que se necesitan. Estas funcionalidades son: garantizar la comunicación con el gestor de procesos de medias para poder codificar el material audiovisual a los formatos ogg y webm, activar la descarga cuando el material se haya codificado y realizar la misma a petición del usuario. Además al estar implementado con una tecnología diferente a la de VideoWeb 1.0, la comunicación se realizaba por servicios web, lo que propiciaba que los tiempos de respuesta fueran muy lentos.

Teniendo en cuenta lo anteriormente planteado se identifica como **problema a resolver** ¿Cómo consultar y recuperar materiales audiovisuales en la plataforma VideoWeb 2.0?

Para dar solución a la problemática planteada, el **objeto de estudio** de la presente investigación son los procesos de búsqueda, visualización, solicitud y descarga de materiales audiovisuales. El **campo de acción** lo constituye los procesos de búsqueda, visualización, solicitud y descarga de materiales audiovisuales en el proyecto Catalogación y publicación de medias.

Planteando así como **objetivo general**: desarrollar un módulo para la consulta y recuperación de materiales audiovisuales en la plataforma VideoWeb 2.0.

Se define como **idea a defender**: la implementación del módulo consulta y recuperación de materiales audiovisuales, garantizará la visualización, solicitud y descarga de las medias en el formato deseado.

Como guía para la realización de esta investigación se proponen las siguientes tareas de la investigación:

- ✚ Caracterización de los procesos de consulta y recuperación de materiales audiovisuales que implementan los sistemas del Departamento de Señales Digitales.
- ✚ Caracterización de las herramientas y tecnologías seleccionadas para la implementación del módulo para la consulta y recuperación de materiales audiovisuales en la plataforma VideoWeb 2.0.

- ✚ Realización del diseño del módulo para la consulta y recuperación para la plataforma VideoWeb 2.0.
- ✚ Implementación del módulo para la consulta y recuperación para la plataforma VideoWeb 2.0.
- ✚ Integración del módulo para la consulta y recuperación de materiales audiovisuales para la plataforma VideoWeb 2.0.
- ✚ Realización de pruebas de caja negra al módulo para la consulta y recuperación para la plataforma VideoWeb 2.0.

Para la realización de estas tareas se emplearon diferentes métodos de la investigación.

Métodos teóricos:

Análítico – sintético: Este método se utilizó con el objetivo de conocer, a través de documentos encontrados durante la investigación la esencia de todo lo relacionado con el proceso de consulta y recuperación de materiales audiovisuales.

Análisis histórico – lógico: Este método permitió realizar un estudio de los sistemas que permiten consultar y recuperar los materiales audiovisuales y así conocer su evolución a lo largo de la historia hasta la actualidad.

Modelación: Mediante este método se realizaron los modelos que especifican cómo se va a desarrollar la aplicación y que permitirán un mejor entendimiento para una posterior implementación.

Principales aportes

Como aporte práctico de la investigación: Se desarrolló un módulo que permite a los usuarios acceder a los materiales audiovisuales digitalizados para visualizarlos, solicitarlos y descargarlos en el formato deseado tales como ogg y webm.

A continuación se muestra la estructura de la investigación que consta de 3 capítulos en los cuales se abordará todo el contenido necesario para la investigación:

Capítulo 1: “Fundamentación Teórica”

Este capítulo está dedicado a los conceptos esenciales relacionados con los procesos de búsqueda, visualización, solicitud y descarga. Además se efectúa un estudio de las soluciones existentes similares al módulo a desarrollar. Describe las tecnologías, los lenguajes de programación y la metodología de desarrollo que se utilizó para la implementación del módulo para la consulta y recuperación de materiales audiovisuales. También se explica la selección y uso de las mismas.

Capítulo 2: “Construcción de la Solución”

En este capítulo se refleja a partir del análisis existente del módulo, una descripción del mismo. Se recoge la propuesta del diseño para la aplicación, el cual incluye diagramas web, así como la descripción de las clases. Además se especifican los patrones de arquitectura y diseño utilizados, conjuntamente se analiza el estándar de codificación.

Capítulo 3: “Prueba”

Este capítulo trata los elementos relacionados con la validación de la solución. Se describen los casos de prueba del sistema y se realizan las pruebas de caja negra sobre las interfaces, detallando los resultados obtenidos en las mismas.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

En este capítulo se plantean conceptos y elementos teóricos que constituyen la base para el desarrollo de esta investigación. Para entender en qué consiste la misma, es necesario conocer todo lo relacionado con su objeto de estudio, en este caso: los procesos de búsqueda, visualización, solicitud y descarga de materiales audiovisuales. Además se analizan otras aplicaciones existentes en el mundo que tienen entre sus funcionalidades algunos de los procesos de búsqueda, visualización, solicitud y descarga de medias. Este capítulo describe la metodología de desarrollo, las herramientas y tecnologías que se van a utilizar para realizar el módulo para la consulta y recuperación de materiales audiovisuales y los lenguajes de programación.

1.1 Términos asociados al problema

1.1.1 Materiales audiovisuales

La definición de materiales audiovisuales procura abarcar el máximo de formas y formatos. Las imágenes en movimiento constituyen la clásica forma de material audiovisual y la forma principal explícitamente mencionada en la Recomendación de la UNESCO de 1980. En realidad incluyen necesariamente también grabaciones sonoras (2). Se define que un material audiovisual es uno de los términos que se utilizan para referirse de forma más generalizada a los audiovisuales, es decir, las imágenes en movimiento, sonidos grabados o ambos a la vez.

1.1.2 Sistemas de búsqueda o recuperación de la información

Búsqueda: Acción de buscar. Investigación, estudio de documentación. (3)

Sistemas de búsqueda o recuperación de la información:

Es el proceso donde se accede a una información previamente almacenada, mediante herramientas informáticas que permiten establecer ecuaciones de búsqueda específicas. Dicha información ha debido de ser estructurada previamente a su almacenamiento (4). Los sistemas de búsqueda no son más que herramientas que permiten procesar consultas realizadas por el usuario para acceder a determinada información que se encuentra almacenada.

1.1.3 Visualización de materiales audiovisuales

Visualización: Acción y resultado de visualizar. (2)

Visualización de Información: El uso de representaciones visuales e interactivas de datos abstractos, soportadas por un medio audiovisual para amplificar el conocimiento. (5)

La visualización de materiales audiovisuales se puede definir como la representación de la información en forma de imágenes o videos que brindan al usuario la posibilidad de conocer dicha información de manera más interactiva.

1.1.4 Solicitud de materiales audiovisuales

Solicitud: Pedir algo con diligencia y cuidado. Documento o memorial en que se solicita algo. (6)

Una solicitud es un pedido que se puede realizar formalmente a una institución u organización. La solicitud de materiales audiovisuales es una petición que realiza el usuario cuando encuentra la media deseada. En dicha solicitud son recogidos los datos del usuario y de los materiales seleccionados por este.

1.1.5 Descarga de materiales audiovisuales

Descargar: Copiar datos (generalmente un archivo entero) de una fuente principal a un dispositivo periférico. El término se utiliza a menudo para describir el proceso de copiar un archivo de un servicio en línea a un ordenador. El término descargar puede también referir a copiar un archivo de un servidor de archivos de red a un ordenador en la red. (7)

La descarga de materiales audiovisuales es el proceso de copiar en un dispositivo periférico o en un ordenador en la red los materiales elegidos por el usuario. Esta operación no puede realizarse sin antes haber realizado la debida solicitud y convertirlos al formato deseado por el usuario.

1.1.6 Conversión de materiales audiovisuales

Conversor: Permite la transformación de los datos de una representación concreta a otra. La conversión de materiales audiovisuales se refiere al proceso de transformar un material, ya sea cambiándole el formato, el tamaño o cualquier otra característica.

1.2 Proceso de búsqueda, visualización, solicitud y descarga de materiales audiovisuales

1.2.1 Descripción General

A partir de la expansión y consolidación de Internet, como medio principal de comunicación electrónica de datos, se ha puesto a disposición de casi todas las personas una importante cantidad de información de

todo tipo. A los efectos de aprovechar todo este potencial de información, es necesario poseer accesos que permitan que la tarea de recuperación sea eficiente y efectiva. La Recuperación de Información (RI) no es un área nueva, pero en la actualidad adquiere un rol más importante debido al valor que tiene la información. Se puede plantear que disponer o no de la información justa en tiempo y forma puede resultar en el éxito o fracaso de una operación. De aquí, la importancia de los Sistemas de Recuperación de Información (SRI) que pueden manejar – con ciertas limitaciones – estas situaciones de manera eficaz y eficiente. (8)

La RI no es más que la recuperación de aquellos documentos que sean relevantes y al mismo tiempo descartar los irrelevantes ante la realización de una consulta por parte de la persona interesada. En la actualidad se encuentra en aumento la demanda de los usuarios para poder buscar, visualizar, solicitar y recuperar materiales audiovisuales. La UCI cuenta con el proyecto VideoWeb perteneciente a la Facultad 6, encargado de gestionar este tipo de contenido. VideoWeb entre sus líneas pretende desarrollar un sistema para la consulta y recuperación de estos materiales, el mismo debe posibilitar además que los usuarios interactúen con los resultados de una búsqueda previamente hecha, permitiéndoles visualizar, solicitar y descargar la media deseada. Este software tiene como objetivo mejorar el proceso de localización, visualización, solicitud y descarga de los audiovisuales en los lugares donde sea desplegado.

1.3 Análisis de otras soluciones existentes

En la actualidad existen diferentes soluciones que son las encargadas de la búsqueda y recuperación de materiales audiovisuales y permiten además visualizarlos y solicitarlos. A continuación se describen las características de algunas de ellas, sus ventajas e inconvenientes con el fin de mejorar la solución que se pretende realizar.

1.3.1 YouTube

Es un sitio web en el cual los usuarios pueden subir y compartir videos. Es la web más popular de video en *streaming*, actualmente ya ofrece videos en formato panorámico, pero la mayoría de los que se suben mantienen la proporción tradicional de 4:3¹. Además del formato panorámico, ahora los videos se pueden subir y ver en alta calidad.

Por lo general, los clips de YouTube son videos cortos caseros, pero muchas discográficas permiten subir

¹ Relación de aspecto de una pantalla de televisión tradicional.

a estas web versiones de baja calidad de los videos musicales más populares. Ofrece videos en 16:9², reproducciones de alta calidad, admite casi cualquier formato de video, garantiza una amplia audiencia, facilita la posibilidad de incrustar los videos en los blog o en las páginas web.

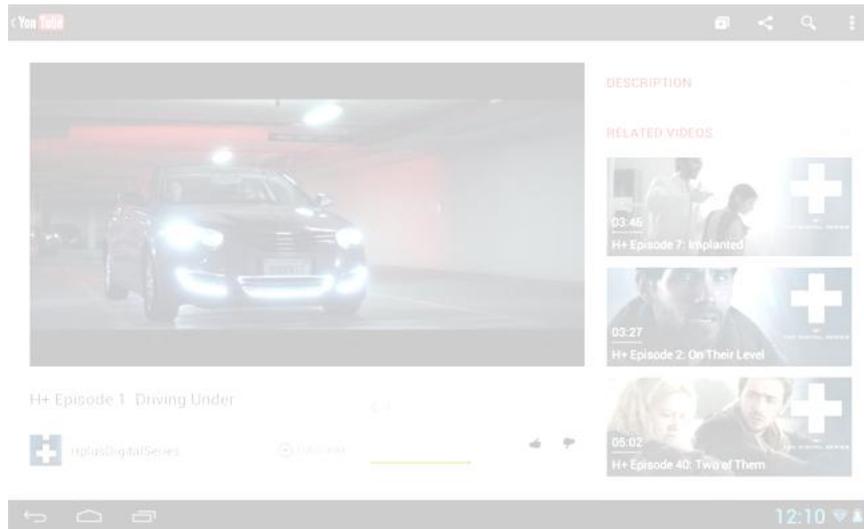


FIGURA 1: INTERFAZ DE YOUTUBE (40)

Al ser un sitio tan popular, surgen algunos inconvenientes. El Spam³ en los comentarios es excesivo y sólo los videos de calidad estándar se pueden incrustar en los blogs o sitios web, además es propietario. (9)

1.3.2 Metacafe

Metacafe tiene un popular sistema de pago por subida de contenido propio, lo que permite a muchos usuarios recibir una compensación por sus videos. Además, debajo de los videos tiene el link para descargarlos y se pueden incrustar en blogs o sitios web. Metacafe tiene una política anti-pornografía aún cuando allí existe una categoría de adultos. La duración de los videos es corta. Permite actualmente la subida de videos con mayor resolución y ahora muchos de los que habían antes se muestran pixelados. Por lo general muchos de los videos son clones de los de YouTube. (9)

² Relación de aspecto usada por la televisión de alta definición.

³ Práctica de enviar indiscriminadamente mensajes de correo electrónico no solicitados.

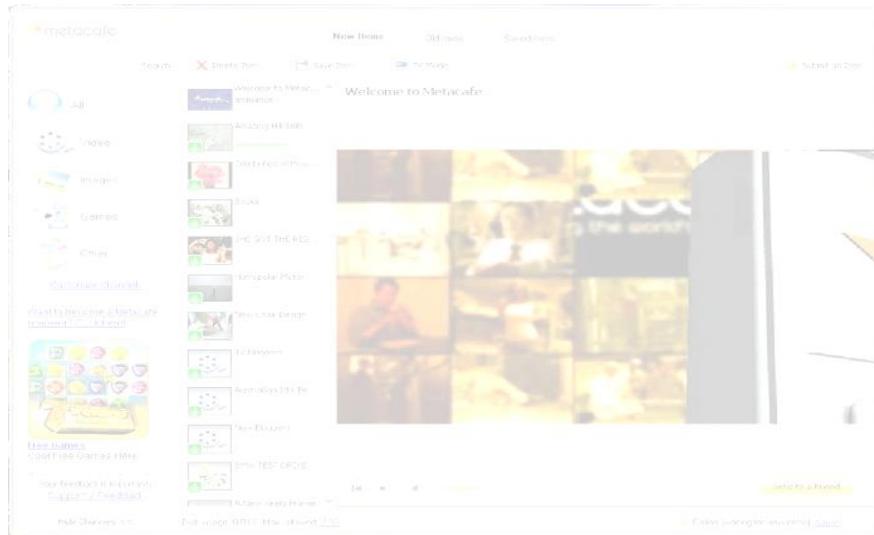


FIGURA 2 : INTERFAZ DE METACAFE (41)

1.3.3 Stage6

Stage6 ofrece un importante salto cualitativo de audio y video además de descarga directa del video y un muy intuitivo buscador. Posee además una interfaz muy amigable que permite continuar viendo el listado de resultados mientras reproduce el video. No tiene límite de subida ni de duración. No es necesario esperar que se haya cargado todo el video, tiene un buffer⁴ que permite visualizar el video mientras se carga. Como negativo se puede destacar que para este tipo de conexiones se necesita un mínimo de ancho de banda de línea de abonado digital asimétrica (ADSL). (10)

⁴ Ubicación de la memoria en un instrumento digital reservada para el almacenamiento temporal de información digital, mientras que está esperando ser procesada.

Además no presenta la opción de descargar los materiales deseados.

Es muy importante la selección de tecnologías y herramientas que se van a utilizar en la realización de un producto de software. A continuación se describe la metodología, los lenguajes de programación y herramientas seleccionadas en el desarrollo del módulo para la consulta y recuperación de materiales audiovisuales.

1.4 Metodología de desarrollo de software

Una metodología de desarrollo de software es un conjunto de pasos y procedimientos que deben seguirse para desarrollar software (11). Pueden estar clasificadas de dos maneras, como metodologías ágiles o robustas. Una cuestión fundamental en el desarrollo de un software es elegir la metodología a usar, pues todo desarrollo de software es riesgoso y difícil de controlar, y si no se utiliza una metodología que guíe el proceso, no se obtiene el resultado deseado y el cliente no queda satisfecho.

Las metodologías pesadas se enfocan en la definición detallada de los procesos y tareas a realizar, herramientas a utilizar, y requiere una extensa documentación, ya que pretende prever todo con anterioridad. Una de las metodologías más conocidas y utilizadas es la Metodología RUP (*Rational Unified Process*). A continuación se expondrán algunas características de la misma.

1.4.1 RUP

RUP es un proceso que define claramente quién, cómo, cuándo y qué debe hacerse. Este aporta herramientas como los casos de uso, que definen los requisitos. Permite la ejecución iterativa del proyecto y del control de riesgos. RUP es un proceso de software genérico que puede ser utilizado para una gran cantidad de tipos de sistemas de software, para diferentes áreas de aplicación, diferentes tipos de organizaciones, diferentes niveles de competencia y diferentes tamaños de proyectos (12). Utiliza el paradigma de orientación a objetos para su descripción. Es un marco de proceso configurable para satisfacer necesidades específicas. Implementa buenas prácticas de desarrollo de software.

Esta metodología divide en 4 fases el desarrollo, estas son:

- ✚ **Inicio:** El objetivo de esta fase es determinar la visión del proyecto.
- ✚ **Elaboración:** El propósito de esta fase es determinar la arquitectura más óptima.
- ✚ **Construcción:** El objetivo de esta fase es obtener la capacidad operacional inicial (construir el software).

- ✚ **Transición:** El objetivo de esta fase es obtener el reléase⁵ del proyecto, dentro de las cuales se realizan varias iteraciones en número variable según el proyecto y en las que se hace un mayor o menor hincapié en las distintas actividades.

RUP divide el proceso de desarrollo en los siguientes flujos de trabajo:

- ✚ **Modelamiento del negocio:** Este flujo se encarga de describir los procesos del negocio.
- ✚ **Requerimientos:** Define qué es lo que el sistema debe hacer.
- ✚ **Análisis y diseño:** Describe cómo el sistema será realizado a partir de la funcionalidad prevista y las restricciones impuestas.
- ✚ **Implementación:** Define cómo se organizan las clases y objetos en componentes, cuáles nodos se utilizarán y la ubicación en ellos de los componentes y la estructura de capas de la aplicación.
- ✚ **Prueba:** Busca los defectos a lo largo del ciclo de vida.
- ✚ **Instalación:** Produce reléase del producto y realiza actividades para entregar el software a los usuarios finales.
- ✚ **Administración del proyecto:** Involucra actividades con las que se busca producir un producto que satisfaga las necesidades de los clientes.
- ✚ **Administración de configuración y cambios:** Describe cómo controlar los elementos producidos por todos los integrantes del equipo de proyecto.
- ✚ **Ambiente:** Contiene actividades que describen los procesos y herramientas que soportarán el equipo de trabajo del proyecto.

Características de RUP

- ✚ **Guiado por Casos de Uso:** ya que el desarrollo está dirigido a satisfacer las necesidades de los usuarios del sistema expresadas en casos de uso. Los casos de uso capturan requisitos funcionales representando piezas de funcionalidad que brindan un resultado de valor al usuario.
- ✚ **Centrado en la Arquitectura:** porque comprende los aspectos estáticos y dinámicos más importantes del sistema. La arquitectura surge de las necesidades de la empresa, como las

⁵ Liberaciones que se le van realizando a un producto determinado. Ejemplo Alfa, Beta.

perciben los usuarios y los inversores y se refleja en los casos de uso.

- ✚ **Iterativo e Incremental:** pues el trabajo se divide en piezas pequeñas o mini proyectos; cada uno proveyendo un subproducto incremental. (13)

Después de realizar este análisis se pudo observar que RUP es muy utilizada para la documentación y realización de software orientados a objetos. El equipo de desarrollo cuenta con experiencia en el uso de esta metodología, aspecto a su favor ya que no requerirá de tiempo para estudiar la misma. Propone varias iteraciones en el proceso de desarrollo del sistema lo que posibilita la corrección de errores y mejoras del software a medida que avanza el proyecto, permitiendo a los desarrolladores poder realizar un producto con mayor calidad. Brinda una abundante documentación, lo que permite un futuro entendimiento de cada componente del producto. Con esta metodología el sistema se va desarrollando y documentando al mismo tiempo, esto es importante ya que, si algún miembro del equipo no puede seguir con el trabajo, el que ocupe su lugar podrá guiarse y conocer qué fue lo que se realizó hasta la fecha.

RUP ayuda a construir un software de alta calidad, desarrollado en el tiempo planificado, permite que se trabaje de forma organizada donde se controle y documente todo lo relacionado con el proyecto, eliminando así los posibles riesgos que se presentan en el mismo, lo cual no podría lograrse sin el empleo de una metodología que se adapte a las características propias del software que se esté desarrollando.

1.5 Herramientas CASE

Se puede definir a las Herramientas CASE (Ingeniería de Software Asistida por Computadora) como un conjunto de programas y ayudas que dan asistencia a los analistas, ingenieros y desarrolladores, durante todos los pasos del ciclo de vida de un software. Permiten integrar análisis de datos y procesos integrados mediante un repositorio, generación de interfaces entre el análisis y el diseño, generación del código a partir del diseño y control de mantenimiento. (14)

1.5.1 Visual Paradigm

Visual Paradigm es una herramienta profesional que permite dibujar todos los tipos de diagramas de clases, generar código desde diagramas y generar documentación. Es muy sencillo de usar, fácil de instalar y actualizar. Otra de las ventajas que posee es que se ajusta al entorno de desarrollo permitiéndoles a los diseñadores, analistas y a todo aquel que trabaje con el mismo, una mayor

organización y claridad en el trabajo. Además presenta soporte multiplataforma. Esta herramienta es bastante común y muy utilizada a nivel mundial por la variedad de funciones que permite realizar, lo que posibilita su utilización para la evolución del proyecto Catalogación y publicación de medias. Algunas ventajas que ofrece para el proyecto son:

- ✚ Diseño centrado en casos de uso y enfocado al negocio, permitiendo generar un diseño de software de gran calidad.
- ✚ Uso de un lenguaje estándar común a todo el equipo de desarrollo facilitando la comunicación.
- ✚ Permite el control de versiones.
- ✚ Modelo y código permanecen sincronizados en todo el ciclo de desarrollo.
- ✚ Permite integrarse con diferentes IDE (entorno de desarrollo integrado). (15)

Visual Paradigm es la herramienta seleccionada ya que emplea UML⁶ para el modelado, es la herramienta por excelencia para ser utilizada en un ambiente de software libre. Permite crear tipos diferentes de diagramas en un ambiente totalmente visual, en este caso facilitará la construcción de los diagramas de clase del diseño, el diagrama de componentes y el diagrama de despliegue logrando que el equipo de desarrollo comprenda mejor el funcionamiento del sistema.

1.6 Tecnologías para el desarrollo web

En los momentos actuales cuando se abordan temas sobre tendencias y tecnologías a utilizar para el desarrollo de algún software o algún componente de software, es importante destacar las relacionadas con el entorno web. Este brinda un grupo de ventajas considerables en cuanto a la disminución de recursos y seguridad informática.

Se puede mencionar también que el entorno web permite tener numerosos clientes independientes de las tecnologías. El capital humano que se necesita para el mantenimiento constante del software disminuye, ya que no es necesaria la instalación del mismo en distintas estaciones, solo se instala en una computadora determinada y se accede a este mediante un navegador web. Debido a esto, la información quedaría almacenada en un solo ordenador y la administración sería de forma centralizada.

Por las características mencionadas anteriormente en los últimos años ha ocurrido un auge considerable en el desarrollo de sitios web para la gestión de información. Todos estos aspectos posibilitaron que se

⁶ Lenguaje Unificado de Modelado.

realizara el módulo para la consulta y recuperación de materiales audiovisuales utilizando tecnologías web.

1.6.1 Modelo Cliente-Servidor

En el mundo de TCP/IP⁷ las comunicaciones entre computadoras se rigen básicamente por lo que se llama modelo Cliente-Servidor, éste es un modelo que intenta proveer usabilidad, flexibilidad, interoperabilidad y escalabilidad en las comunicaciones. Desde el punto de vista funcional, se puede definir el modelo Cliente-Servidor como una arquitectura distribuida que permite a los usuarios finales obtener acceso a la información en forma transparente aún en entornos multiplataforma. En el modelo cliente servidor, el cliente envía un mensaje solicitando un determinado servicio a un servidor (hace una petición), y este envía uno o varios mensajes con la respuesta (provee el servicio).

Las características básicas de la arquitectura Cliente-Servidor:

- ✚ Se establece una relación entre procesos distintos, los cuales pueden ser ejecutados en la misma máquina o en máquinas diferentes distribuidas a lo largo de la red.
- ✚ El ambiente es heterogéneo. La plataforma de hardware y el sistema operativo del cliente y del servidor no son siempre la misma. Precisamente una de las principales ventajas de esta arquitectura es la posibilidad de conectar clientes y servidores independientemente de sus plataformas.
- ✚ Un servidor ofrece servicio a variados clientes de forma concurrente.
- ✚ Si ocurrieran modificaciones en el servidor estas afectarían en ninguna o poca escala al cliente.

(16)

Para el desarrollo del sistema una de las ventajas fundamentales que brinda esta tecnología es la de ser una infraestructura versátil, modular y basada en mensajes que mejora la portabilidad, la interoperabilidad y la escalabilidad de los sistemas. Posibilita que los usuarios se conecten de manera remota a la aplicación, además con el uso de este modelo aumenta la interactividad y se facilita el mantenimiento de las aplicaciones, por estas razones es que se decidió utilizarlo en el proyecto Catalogación y publicación de medias.

⁷ : Protocolo de Control de Transmisión (TCP) y Protocolo de Internet (IP).

1.7 Lenguaje de programación

Un lenguaje de programación es un lenguaje diseñado para describir el conjunto de acciones consecutivas que un equipo debe ejecutar. Por lo tanto, un lenguaje de programación es un modo práctico para que los seres humanos puedan dar instrucciones a un equipo. (17)

El lenguaje de programación es el instrumento que permite al programador elegir los datos que se deben usar, cómo serán guardados, cómo se presentarán al usuario final, qué tipo de actuaciones se deben tomar bajo determinadas condiciones. Para lo cual los lenguajes de programación intentan estar lo más cercano posible al lenguaje humano o natural. Los lenguajes de programación web facilitan el trabajo a los desarrolladores de aplicaciones. Estos se clasifican en lenguajes del lado cliente y lenguajes del lado servidor.

1.7.1 Lenguaje del lado del cliente

Un lenguaje del lado cliente es totalmente independiente del servidor, lo cual permite que la página pueda ser albergada en cualquier sitio. Pero esta página no se verá bien si la computadora cliente no tiene instalados los *plugins* adecuados. El código, tanto del hipertexto como de los scripts, es accesible a cualquiera y por ello puede afectar a la seguridad. (18)

Lenguaje de Marcado de Hipertexto (HTML)

El Lenguaje de Marcado de Hipertexto (HTML) es un lenguaje de marcas hipertextuales, un lenguaje diseñado para estructurar textos para generar páginas web. Gracias a Internet y a los navegadores web, el HTML se ha convertido en el formato más fácil para la creación de páginas web debido a su sencillez.

La mayoría de las etiquetas del lenguaje HTML son semánticas. La interpretación de las etiquetas es realizada por el navegador web. El lenguaje HTML es extensible, se le pueden añadir características, etiquetas y funciones adicionales para el diseño de páginas web. (19) Haciendo uso de este lenguaje se llevará a cabo el desarrollo del reproductor, garantizando que los usuarios puedan observar las medias seleccionadas, también posibilitará una mejor organización y estructura del contenido. Este lenguaje permite generar un producto vistoso, rápido y sencillo.

JavaScript

JavaScript es un lenguaje interpretado en el cliente por el navegador al momento de cargarse la página, es multiplataforma, cuyo código se incluye directamente en el mismo documento HTML. Es utilizado para

crear pequeños programas encargados de realizar acciones dentro del ámbito de una página web. Su uso se basa fundamentalmente en la creación de efectos especiales en las páginas y la definición de interactividades con el usuario.

Tiene como características principales las siguientes:

- ✚ Es interpretado por el cliente.
- ✚ Está basado en objetos. No es un lenguaje de programación orientada a objetos (POO). JavaScript no emplea clases ni herencia, típicas de la POO.
- ✚ Su código se integra en las páginas HTML, incluido en las propias páginas. (20)

JavaScript permite la creación de funciones que garantizan la correcta entrada de los datos por parte del usuario, además hace posible que se le muestren a este los datos de forma más comprensible.

JQuery

JQuery es una biblioteca de clases de JavaScript que permite realizar modificaciones sobre el contenido de una página web de una manera sencilla e independiente del navegador. JQuery es un software libre y de código abierto, ofrece una serie de funcionalidades basadas en JavaScript que de otra manera requerirían de mucho más código, es decir, con las funciones propias de esta biblioteca se logran grandes resultados en menos tiempo y espacio. La característica principal de la biblioteca es que permite cambiar el contenido de una página web sin necesidad de recargarla. (21)

Dicha biblioteca permite la creación de pestañas dinámicamente, logrando una mejor visualización del contenido, contribuirá a la construcción de una barra de progreso permitiendo conocer el estado de conversión de una media. Además JQuery será utilizado para la comunicación cliente-servidor posibilitando cargar el contenido de las páginas dinámicamente.

1.7.2 Lenguaje del lado del servidor

La programación del lado del servidor es una tecnología que consiste en el procesamiento de una petición de un usuario mediante la interpretación de un script en el servidor web para generar páginas HTML dinámicamente como respuesta(22).Un lenguaje del lado servidor es independiente del navegador utilizado, y no necesita *plugins* especiales para visualizar correctamente cualquier página.

Hypertext Pre-processor (PHP)

PHP es un lenguaje de script interpretado en el lado del servidor utilizado para la generación de páginas web dinámicas, embebido en páginas HTML y ejecutado en el servidor. La meta del lenguaje es permitir rápidamente a los desarrolladores la generación dinámica de páginas.

Es un lenguaje multiplataforma. Completamente orientado al desarrollo de aplicaciones web dinámicas con acceso a información almacenada en una base de datos. El código fuente escrito en PHP es invisible al navegador y al cliente ya que es el servidor el que se encarga de ejecutarlo y enviar su resultado HTML al navegador, esto hace que la programación en PHP sea segura y confiable. Tiene capacidad de conexión con la mayoría de los motores de base de datos que se utilizan en la actualidad, destaca su conectividad con MySQL y PostgreSQL. Es libre, por lo que se presenta como una alternativa de fácil acceso para todos. (23)

Es un lenguaje de programación gratuito e independiente de plataforma, rápido, con una gran biblioteca de funciones y mucha documentación donde se pueden encontrar ejemplos de cada una de ellas. Es usado para el procesamiento de las funcionalidades, es decir mediante este es posible llevar a cabo la implementación de las operaciones del sistema. Este lenguaje permite la validación de datos por el lado del servidor, y garantiza la seguridad del sistema definiendo a través de él los roles que tendrán acceso al mismo.

1.8 Sistema de gestión de contenido (CMS)

Un sistema de gestión de contenido (CMS) es un software que permite crear una estructura de soporte (*Framework*) para la creación y administración de contenidos, principalmente en páginas web, por parte de los usuarios del mismo. Consiste en una interfaz que controla una o varias bases de datos donde se aloja el contenido del sitio. El sistema permite manejar de manera independiente el contenido y el diseño. Así, es posible manejar el contenido y darle, en cualquier momento, un diseño distinto al sitio sin tener que darle formato al contenido de nuevo, además de permitir la fácil y controlada publicación en el sitio a varios editores. (24).

1.8.1 Drupal 7

Drupal, es una herramienta usada para la gestión de contenido modular muy configurable que permite publicar artículos, imágenes, u otros. Es un CMF⁸ libre, escrito en PHP, desarrollado y mantenido por una activa comunidad de usuarios. 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.

⁸ Marco de gestión de contenidos, que permite personalizar la creación de un sistema de gestión de contenidos.

Su flexibilidad y adaptabilidad, así como la gran cantidad de módulos adicionales disponibles, hace que sea adecuado para realizar muchos tipos diferentes de sitios web. Puede usarse para aplicaciones de varios tipos como pueden ser: portales comunitarios, foros de discusión, sitios web corporativos, aplicaciones de Intranet, sitios personales, aplicaciones de comercio electrónico, directorio de recursos y sitios de redes sociales (25).

La versión 7 de Drupal introduce las últimas tecnologías web y mejora notablemente la experiencia de usuario. Angie Byron (también conocida como Webchick), mantenedora del núcleo de Drupal 7 dice que: *“Drupal 7 pone en juego una nueva y mejorada interfaz de usuario, mejoras de accesibilidad, una mejor gestión de imágenes, soporte para metadatos de enriquecimiento del contenido, mejoras de seguridad, escalabilidad y un aumento de la interoperabilidad con las bases de datos, un juego de 30.000 test automatizados para asegurar la estabilidad, y mucho más.”*(26)

Además la versión 6 de Drupal según el sitio Drupal.org dejará de tener soporte por la comunidad de desarrollo dentro de poco tiempo. Esta es otra de las razones para migrar a Drupal7.

1.9 Entorno de Desarrollo Integrado (IDE)

Un entorno de desarrollo integrado (*Integrated Development Environment* o IDE) es un programa compuesto por una serie de herramientas que utilizan los programadores para desarrollar código. Esta herramienta puede estar pensada para su utilización con un único lenguaje de programación o bien puede dar cabida a varios de estos.

1.9.1 Netbeans

Netbeans es un cómodo entorno de desarrollo integrado, pensado para facilitar las tareas de los desarrolladores de aplicaciones, ofreciendo un entorno completamente gráfico. Entre las principales características de Netbeans IDE, se tiene su enorme versatilidad, lo que permite realizar programas en diferentes lenguajes de programación, como C/C++, Ruby, Ajax, PHP, JavaScript, entre otros.

El IDE Netbeans cuenta con una enorme cantidad de extensiones y *plugins*, que pueden incorporarse al sistema para facilitar las tareas de programación. También, se puede destacar su intuitiva interfaz de usuario, su sistema de depuración de programas y las demás herramientas de ayuda, que facilitan el desarrollo de software. (27)

Netbeans posee un fuerte editor de código fuente, sobre todo el de PHP, ya que provee todas las ventajas que presenta este lenguaje, además es gratuito y de código abierto. Presenta todas las herramientas para no solo elaborar aplicaciones en entorno web sino también de escritorio y móviles. Este entorno de desarrollo cuenta con un buen completamiento de código de los lenguajes de programación y se integra con múltiples sistemas de control de versiones lo que facilita el trabajo para los desarrolladores. Todas esas características posibilitaron que el proyecto Catalogación y publicación de medias escogiera a Netbeans como IDE de desarrollo.

1.10 Sistemas Gestores de Bases de Datos (SGBD)

Se trata de un conjunto de programas no visibles al usuario final que se encargan de la privacidad, la integridad, la seguridad de los datos y la interacción con el sistema operativo. Proporciona una interfaz entre los datos, los programas que los manejan y los usuarios finales. (28)

Las operaciones que el usuario hace contra la base de datos están controladas por el gestor. Este almacena una descripción de datos, así como los usuarios permitidos y los permisos. Además debe haber un usuario administrador encargado de centralizar todas estas tareas.

Hoy en día existen numerosos SGBD que tienen características propias, no obstante todos deben poseer los siguientes aspectos: abstracción de la información, la independencia de los datos, redundancia mínima, consistencia en los datos, seguridad, integridad, respaldo y recuperación, tiempo de respuesta y control de concurrencia.

1.10.1 PostgreSQL

PostgreSQL es un sistema gestor de base de datos relacionales orientado a objeto. Es el sistema de gestión de bases de datos de código abierto más avanzado del mundo, utiliza un modelo cliente-servidor y usa multiprocesos en vez de multihilos para garantizar la estabilidad del sistema. Un fallo en uno de los procesos no afectará el resto y el sistema continuará funcionando. Se ejecuta en casi todos los principales sistemas operativos: Linux, Unix, BSDs, Mac OS, Beos, Windows. Tiene documentación muy bien organizada, pública y libre, con comentarios de los propios usuarios. Es altamente adaptable a las necesidades del cliente. (29)

Trabajar con este gestor es factible ya que es muy rápido, tiene buenas utilidades de administración, es confiable y fácil de usar tanto para volúmenes de datos grandes como pequeños. PostgreSQL hoy en día

ofrece un amplio y muy útil conjunto de funciones.

Además la conectividad, velocidad y seguridad, lo hacen altamente conveniente ya que tiene mejor control de acceso a usuarios. Al mismo tiempo, los lenguajes que se proponen para el desarrollo de la plataforma logran una buena interacción con este gestor lo que resulta compatible y de mucho beneficio a la hora de diseñar de forma rápida y eficaz aplicaciones dirigidas a bases de datos.

1.11 Conclusiones Parciales

En este capítulo del estudio realizado sobre los principales conceptos y términos asociados se obtuvo un mejor entendimiento del tema a tratar. Además se analizaron algunos sistemas similares, lo que permitió identificar algunas características de impacto para una solución de este tipo, como una interfaz intuitiva, comprensible y amigable, aspecto de gran importancia para garantizar un mejor resultado. A partir del estudio realizado se detectó que ninguna de las soluciones puede ser utilizada por su forma de integración, además no tienen una interfaz de comunicación para poder utilizar sus funcionalidades desde otro sistema como el que se desea implementar. Se seleccionaron las herramientas y tecnologías a emplear en el desarrollo del producto, con el objetivo de lograr un software de calidad y altamente funcional. El uso de la metodología RUP, el IDE de desarrollo Netbeans, el CMF Drupal y el lenguaje de programación PHP, además de JQuery como biblioteca de JavaScript en conjunto con HTML posibilitaron que el producto fuera desarrollado en el tiempo establecido y con la documentación necesaria permitiendo el futuro entendimiento del mismo.

CAPÍTULO 2: CONSTRUCCIÓN DE LA SOLUCIÓN

La implementación del módulo de consulta y recuperación de materiales audiovisuales para el proyecto Catalogación y publicación de medias, está fundamentada en un grupo de requisitos funcionales realizados con anterioridad por dicho proyecto. Dichos requisitos y sus descripciones se podrán consultar en el anexo 1 del presente documento. En este capítulo se recoge el diseño del módulo para la consulta y recuperación de materiales audiovisuales que se propone, el cual incluye diagramas web, así como la descripción de las clases. Además se especifican los patrones de diseño utilizados y el estándar de codificación manejado en la aplicación.

2.1 Descripción del sistema

El flujo de información que informatiza el módulo para la consulta y recuperación de materiales audiovisuales comienza cuando el usuario realiza una búsqueda de las medias deseadas, una vez realizada la misma se mostrarán los resultados obtenidos los cuales podrán ser visualizados y en caso deseado solicitados. Cuando se realiza una solicitud esta puede ser de tipo interna o externa. Las solicitudes internas recogerán los datos del usuario que la realizó. En caso de que la solicitud realizada sea externa se recogerán los datos del cliente que efectuó dicho pedido, en ambos casos las solicitudes guardarán los datos de la media seleccionada así como el formato al cual desean convertir el material. Una vez realizada la solicitud, el módulo se comunicará con el gestor de procesos de media el cual es el encargado de codificar el material al formato seleccionado. El flujo termina cuando la media finaliza el proceso de conversión y es descargada por el usuario.

El sistema brinda la posibilidad de que los usuarios se autentiquen para poder realizar las funcionalidades propias del mismo, un usuario autenticado podrá buscar las medias deseadas, cuando las encuentre podrá reproducirlas y realizar la solicitud de las mismas. Por otra parte el usuario que brinda atención al público además de realizar las funcionalidades antes mencionadas podrá atender las solicitudes y descargar las medias cuando estén listas. Ambos usuarios podrán cerrar sesión una vez terminen de usar el sistema. Para una mejor comprensión se recomienda ver el diagrama de caso de uso del sistema en el Anexo 2.

2.2 Diseño de la Base de Datos

El modelo de datos es una representación abstracta de los datos de una organización y las relaciones entre ellos. El propósito de un modelo de datos es, por una parte, representar los datos y, por otra, ser comprensible. (30)

Un modelo de datos es un conjunto de conceptos que permiten describir los datos y las relaciones que existen entre ellos. Elaborar un correcto modelo de datos es imprescindible para manejar y obtener correctamente los datos.

En la Figura 4 que se muestra a continuación aparece una porción del modelo de datos, que representa físicamente la base de datos. Esta fue diseñada en el proyecto Catalogación y publicación de medias para almacenar y gestionar la información referente a todo el sistema. En la misma se reflejan las tablas relacionadas con el proceso de consulta y recuperación de materiales audiovisuales que se implementa en el módulo con este propósito.

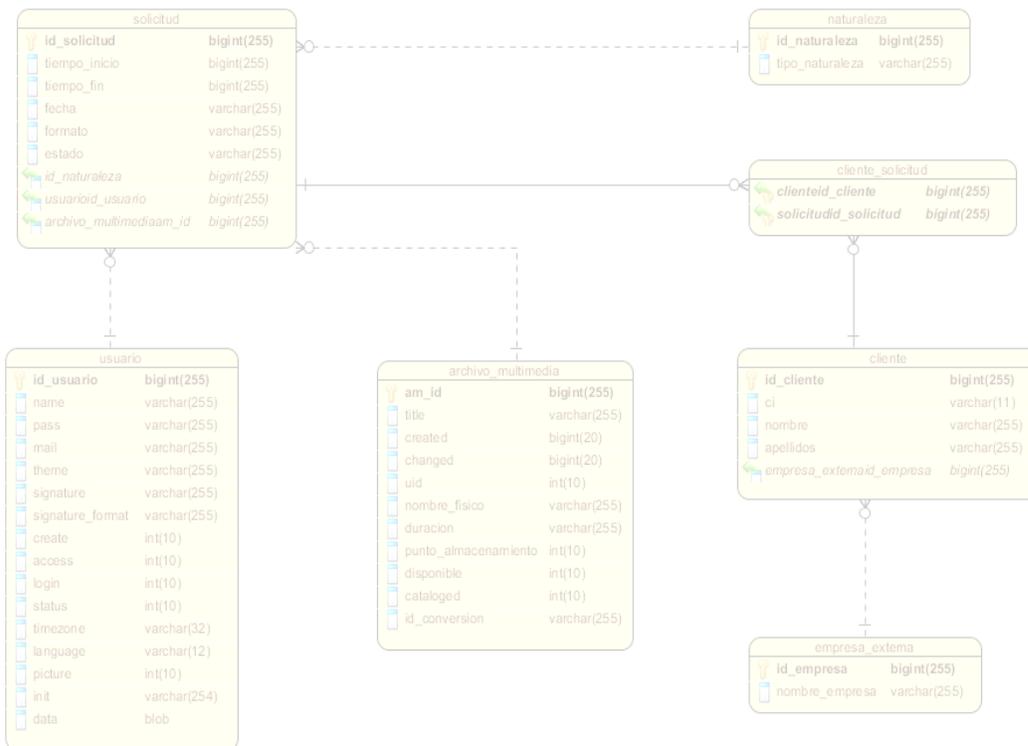


FIGURA 4: MODELO DE DATOS

Entre las tablas existentes en la base de datos, destinadas a manipular la información de la recuperación de los materiales audiovisuales, se encuentran *Naturaleza*, *Cliente*, *Solicitud*, *Empresa_Externa* y *Archivo_Multimedia* estas son muy importantes en el proceso de solicitud y descarga de las medias. Con *Archivo_Multimedia* se gestionan los datos propios de los archivos de video como el título y la duración. *Naturaleza* se refiere al tipo de solicitud ya sea externa o interna. La tabla *Cliente* es la encargada de almacenar la información relacionada con los clientes que realizan solicitudes externas como son el carnet de identidad, el nombre, los apellidos y la entidad a la que pertenece. La tabla *Empresa_Externa* guarda el nombre de la empresa a la que pertenece el cliente. Por su parte la tabla *Solicitud* recoge los datos pertenecientes a la solicitud que realizó el usuario, como la fecha, el tiempo de inicio, el tiempo de fin, el formato y el estado (codificando, pendiente, lista).

2.3 Patrones Arquitectónicos

Los patrones arquitectónicos se encargan de definir la estructura de un sistema, estos a su vez se componen de subsistemas con sus responsabilidades, también tienen una serie de directivas para organizar los componentes del mismo sistema, con el objetivo de facilitar la tarea del diseño de tal sistema. Un patrón arquitectónico se enfoca a dar solución a un problema en específico. (31)

El concepto de arquitectura de software se refiere a la estructuración del sistema que se crea en etapas tempranas del desarrollo. Esta estructuración representa un diseño de alto nivel del sistema que tiene dos propósitos primarios: satisfacer los atributos de calidad (desempeño, seguridad, modificabilidad), y servir como guía en el desarrollo.

De acuerdo al Software Engineering Institute (SEI), la Arquitectura de Software se refiere a las estructuras de un sistema, compuestas de elementos con propiedades visibles de forma externa y las relaciones que existen entre ellos(32). En el software se considera que un estilo de arquitectura, es una definición de tipos particulares de componentes, así como de la interacción entre ellos los cuales satisfacen un conjunto de restricciones(33). Para modelar la arquitectura del sistema se utilizó el patrón arquitectónico Modelo-Vista-Controlador (MVC).

2.3.1 Modelo-Vista-Controlador

El patrón arquitectónico que se utilizó para la realización de este sistema es el Modelo-Vista-Controlador, este separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos. Además provee simplicidad en el mantenimiento del sistema.

El patrón MVC se ve frecuentemente en aplicaciones web, donde la vista es la página HTML que provee de datos dinámicos a la página, el modelo es el Sistema de Gestión de Base de Datos y la lógica de negocio o controlador es el responsable de recibir los eventos de entrada desde la vista.

Los elementos del patrón son los siguientes:

El modelo es el responsable de: Acceder a la capa de almacenamiento de datos. Define las reglas de negocio (la funcionalidad del sistema). Lleva un registro de las vistas y controladores del sistema.

El controlador es el responsable de: Recibir los eventos de entrada (un clic, un cambio en un campo de texto). Contiene reglas de gestión de eventos.

Las vistas son responsables de: Recibir datos del modelo y mostrarlos al usuario. Tienen un registro de su controlador asociado.

2.3.2 Patrones de diseño de clases

Los patrones de diseño de clases son el esqueleto de las soluciones a problemas comunes en el desarrollo de software (34). En otras palabras, brindan una solución ya probada y documentada a problemas de desarrollo de software que están sujetos a contextos similares.

Los patrones de diseño pretenden evitar la reiteración en la búsqueda de soluciones a problemas ya conocidos y solucionados anteriormente, formalizar un vocabulario común entre diseñadores además de facilitar el aprendizaje de las nuevas generaciones de diseñadores condensando conocimiento ya existente (35). Para dar solución al problema planteado se hizo necesaria la utilización de algunos de ellos.

Experto

Lo que plantea el patrón experto es asignar una responsabilidad al experto en información, es decir, la clase que tiene la información necesaria para cumplir con la responsabilidad (35). Un ejemplo donde se presencia la utilización de este patrón es en la clase *recuperacion_controladora* la cual tiene la información referente a las solicitudes y por lo tanto las funcionalidades para manipularlas (modificarlos o eliminarlos) están implementadas en esta clase.

Creador

Este patrón tiene como objetivo principal asignar a las clases la responsabilidad de crear una instancia de otra, soportando mayor claridad en el código y posibilitando el encapsulamiento y la reusabilidad (35). Este patrón se evidencia en la clase *recuperacion_controladora* que es la encargada de instanciar un objeto de la clase *recuperacion_modelo* para instanciar las funcionalidades que posee esta clase.

Bajo Acoplamiento

El patrón bajo acoplamiento se utilizó para establecer un uso mínimo de clases entrelazadas de modo que si se hacen modificaciones solamente afecten a la menor cantidad de clases (35). Esto se evidencia en la estructura de componentes que se ha empleado, como el reproductor, el componente de búsqueda, y las solicitudes, los cuales pueden ser reutilizados de manera independiente sin que sus funcionalidades se vean afectadas por estar separados del resto de la estructura del módulo.

Alta Cohesión

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 (35). Dicho patrón se pone de manifiesto en la propia arquitectura definida para el sistema ya que cada clase tiene las funcionalidades que ellas implementan sin embargo colaboran entre sí para realizar sus tareas.

Controlador

El patrón controlador se utilizó pues asigna la responsabilidad del manejo de un mensaje de los eventos de un sistema a una clase que represente el sistema global (36). En la implementación del módulo se puede ver que la clase controladora *recuperacion_controladora* se encarga de encausar las funcionalidades realizadas en el sistema.

Observador

El patrón observador define una dependencia de uno a muchos entre objetos, de forma que cuando un objeto cambia de estado se notifica y actualizan automáticamente todos los objetos. Los observadores necesitan saber cuándo se produce un cambio en el objeto observado (35). La solución usando el patrón observador trata de que el objeto observador que en este caso es el componente (*recuperacion_materiales_audiovisuales*) va a estar observando al objeto observado

(*resultado_busqueda*), que se encuentra en el componente de búsqueda, una vez que el objeto observado cambie, el observador notificará el cambio y comenzará a manejar los nuevos datos.

Instancia única (*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 (35). 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.

2.4 Diagramas de clases del diseño

Los diagramas de clases del diseño constituyen un elemento esencial en la concepción de la aplicación que se propone, ya que servirán de guía a los desarrolladores para la construcción del sistema que se desea implementar, contribuyendo así a una mayor calidad del producto final. Además describe gráficamente las especificaciones de las clases de software y de las interfaces en una aplicación. A continuación se muestran en la Figura 5,6 y 7 los diagramas de clases del diseño correspondientes a cada caso de uso.

El Caso de Uso *Gestionar_Solicitud* (ver descripción en el anexo 3) describe el proceso de respuesta del sistema ante la petición de un usuario de insertar, modificar o eliminar una solicitud. Las principales clases que intervienen en dicho caso de uso son:

- ✚ ***recuperacion_vista***: en esta clase el método ***Form_Datos_Solicitud ()*** se encarga de construir la vista que contendrá el formulario que ofrece los campos para insertar los datos de la solicitud.
- ✚ ***recuperacion_controladora***: en esta clase está implementado el método ***Gestionar_Solicitud ()*** encargado de insertar, modificar o eliminar las solicitudes.
- ✚ ***recuperacion_modelo***: esta clase se encarga del acceso a los datos de las solicitudes a través de los siguientes métodos: ***DB_Agregar_Solicitud ()***, ***DB_Modificar_Solicitud ()*** y ***DB_Eliminar_Solicitud ()***.

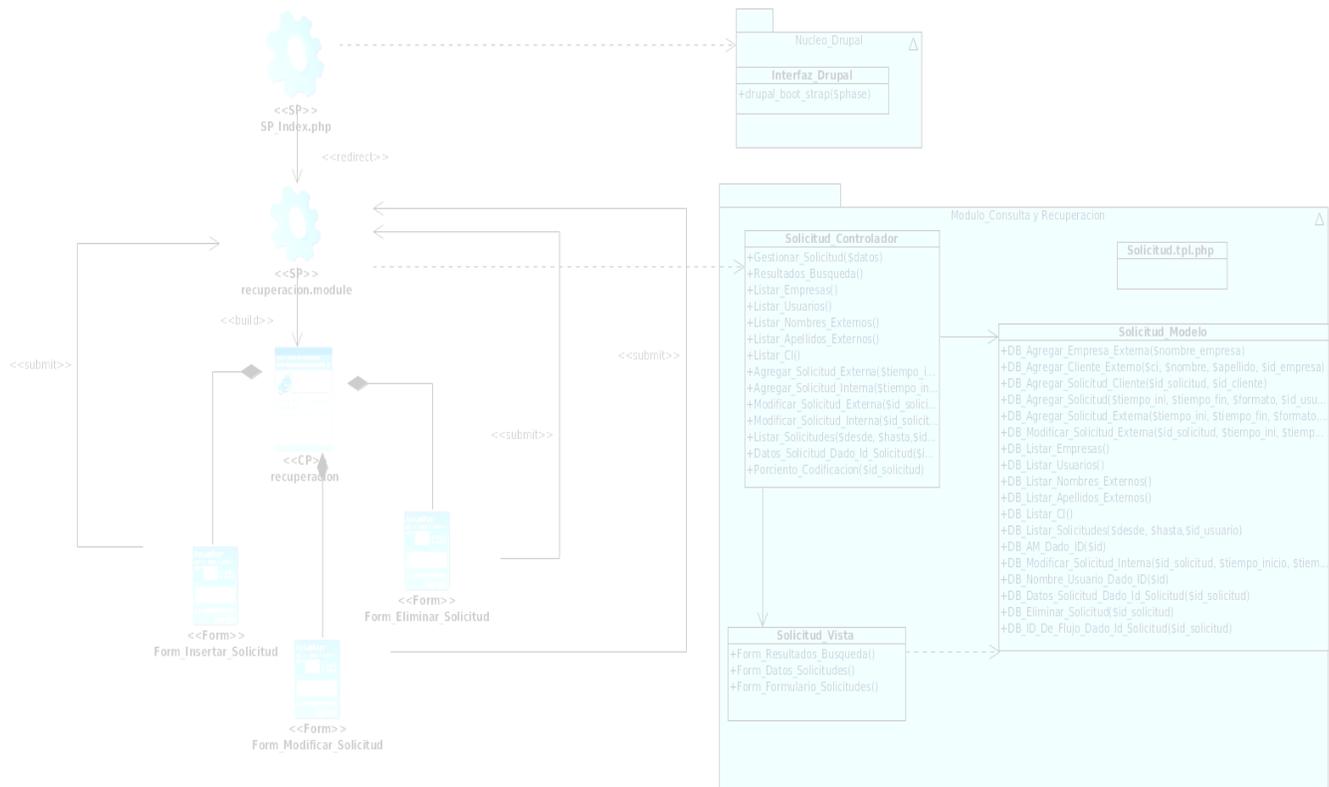


FIGURA 5: DIAGRAMA DE CLASES DEL DISEÑO GESTIONAR SOLICITUD

El Caso de Uso *Atender_Solicitud* (ver descripción en el anexo 3) permite conocer si la solicitud está siendo atendida o no. Este resultado es mostrado en un formulario con la lista de las solicitudes donde se muestra además el estado en que se encuentran las mismas (en espera, codificando o listo). Las principales clases que intervienen en este caso de uso son:

- ✚ **recuperacion_vista:** esta clase contiene el método **Form_Formulario_Solicitudes ()** encargado de mostrar las listas de solicitudes realizadas por el usuario.
- ✚ **recuperacion_controladora:** en esta clase se encuentra la funcionalidad **Listar_Solicitudes ()** que tiene como objetivo listar todas las solicitudes realizadas por el usuario y el método **Porcentaje_Codificacion ()** encargado de ir chequeando en qué porcentaje de conversión se encuentra la solicitud.
- ✚ **recuperacion_modelo:** clase que contiene el método **DB_Listar_Solicitudes ()** encargado de buscar en la base de datos las solicitudes realizadas.

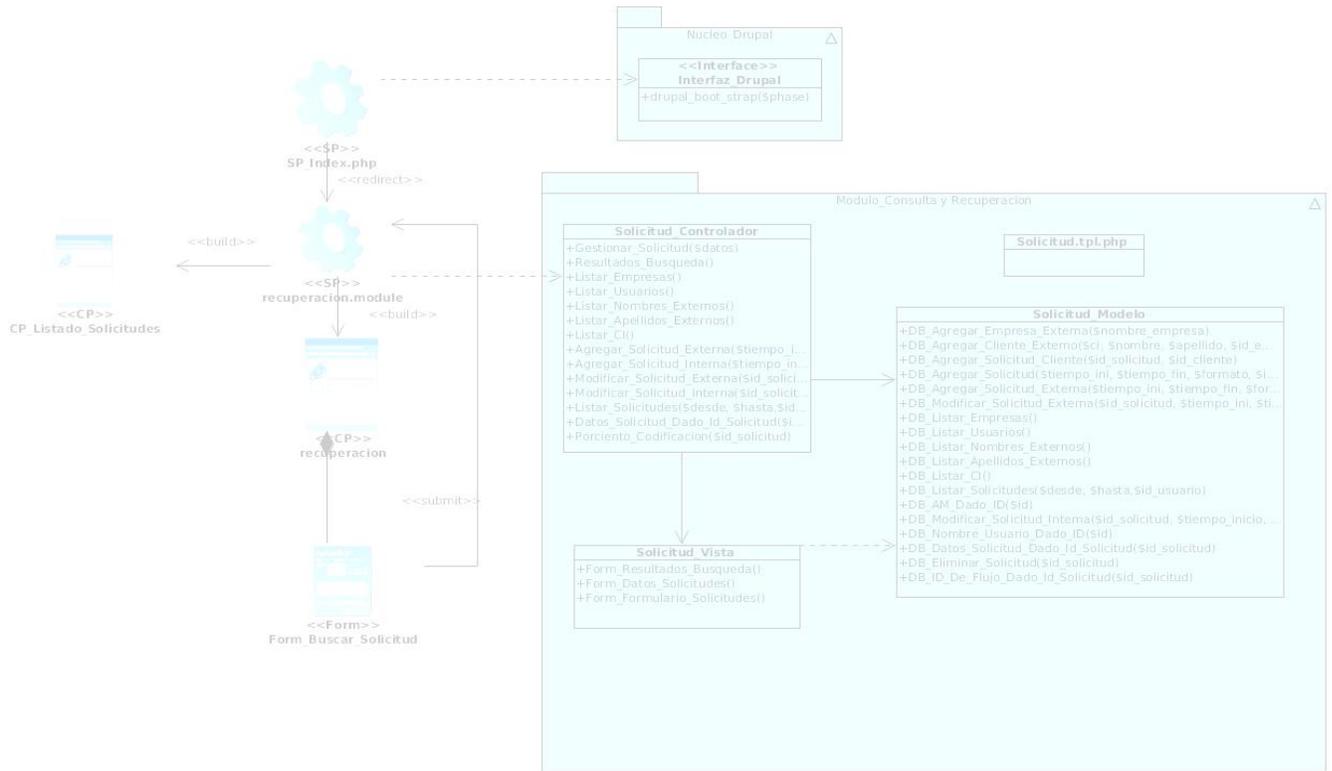


FIGURA 6: DIAGRAMA DE CLASES DEL DISEÑO ATENDER SOLICITUD.

El caso de uso *Descargar_Material* (ver descripción en el anexo 3) permite que los usuarios descarguen la media solicitada en el formato deseado, esta operación se realiza una vez que el estado de la solicitud sea listo. En este caso de uso una de las clases principales es:

- ✚ **recuperacion_vista:** en esta clase se ubica el método encargado de descargar la media una vez terminada la conversión. La funcionalidad encargada de realizar este proceso es **Descargar_Material ()**.

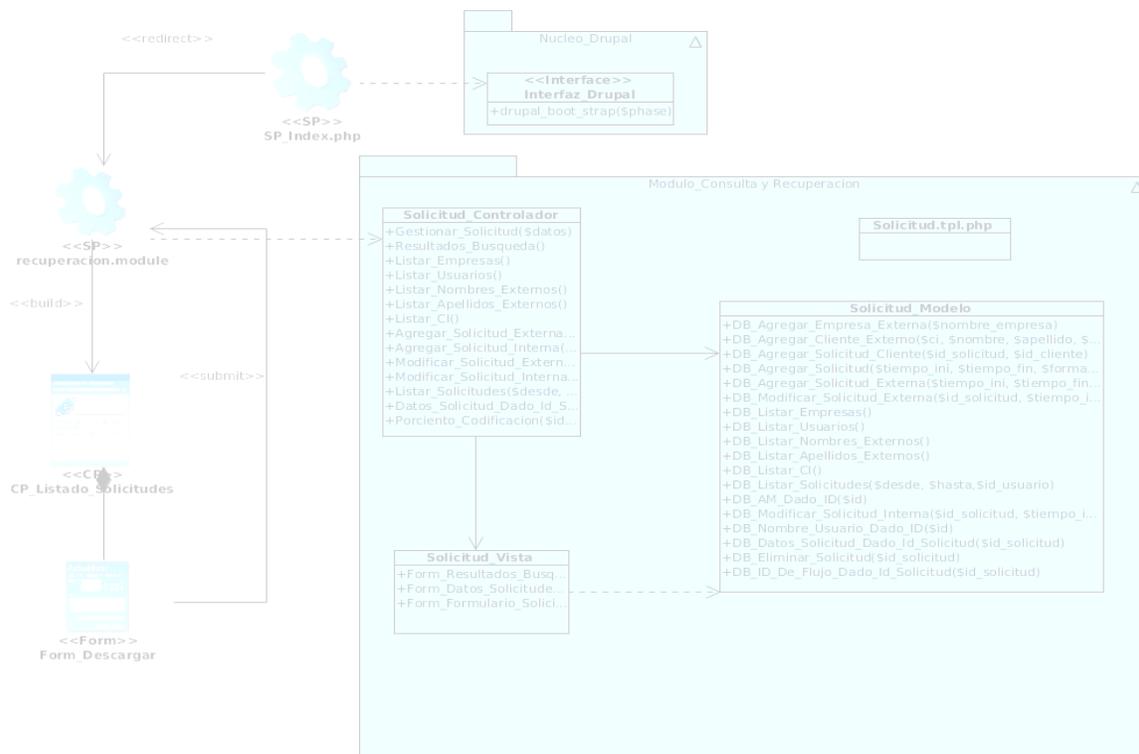


FIGURA 7: DIAGRAMA DE CLASES DEL DISEÑO DESCARGAR MATERIAL

2.5 Diagrama de Implementación

Un diagrama de implementación muestra la estructura del código (Diagrama de componentes) y la estructura del sistema en ejecución. (36)

Un diagrama de componentes muestra las dependencias lógicas entre componentes software, sean estos componentes fuentes, binarios o ejecutables. En la Figura 8 se muestra el diagrama de componente referente al módulo de consulta y recuperación de materiales audiovisuales.

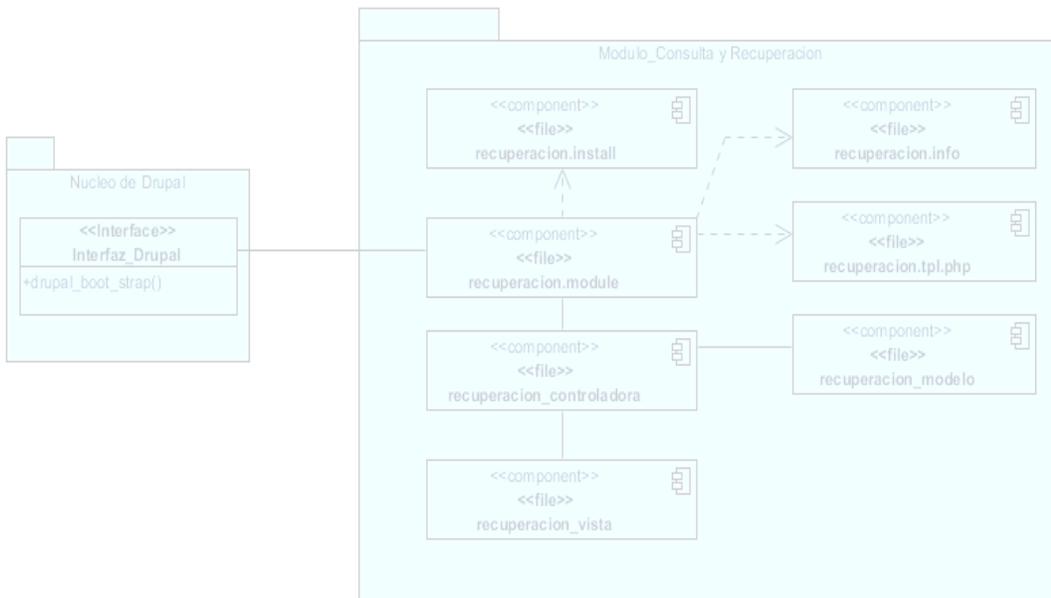


FIGURA 8: DIAGRAMA DE COMPONENTES

La imagen anterior muestra la distribución e interacción entre los componentes que conforman el módulo web de consulta y recuperación de materiales audiovisuales, teniendo en cuenta el diseño y la arquitectura establecida para el mismo.

2.6 Diagrama de Despliegue

Los Diagramas de Despliegue muestran las relaciones físicas de los distintos nodos que componen un sistema y el reparto de los componentes sobre dichos nodos. Estos representan a los nodos y sus relaciones. Los nodos son conectados por asociaciones de comunicación tales como enlaces de red, conexiones TCP/IP (37). La Figura 9 muestra la distribución entre los nodos que componen el sistema.

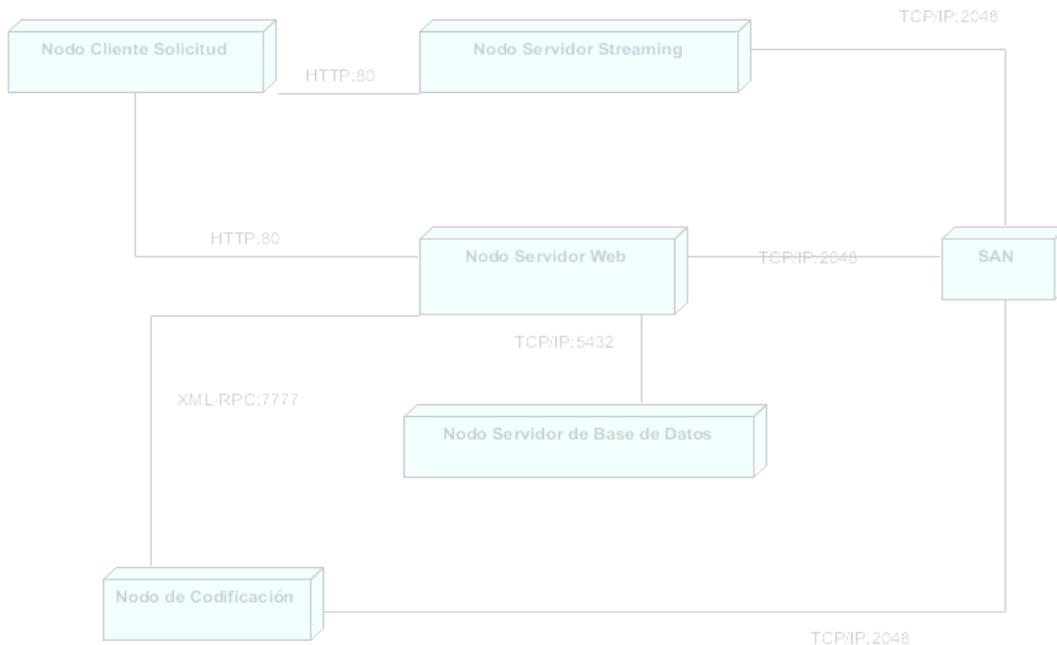


FIGURA 94: DIAGRAMA DE DESPLIEGUE

El nodo Cliente Solicitud representa las computadoras que permitirán la interacción de los usuarios con la aplicación cliente del sistema, brindando las funcionalidades de consulta y recuperación de materiales audiovisuales. Este se conecta con los nodos donde se encuentran el Servidor Web y el Servidor streaming con el protocolo HTTP. El Servidor Web a su vez se comunica con el Servidor de Base de Datos mediante el protocolo TCP/IP y con el nodo de Codificación por el protocolo XML-RPC encargado de convertir la media al formato seleccionado. Tanto el nodo Servidor streaming, el Servidor Web como el nodo de Codificación se comunican a través del protocolo TCP/IP por el puerto 2048 con el nodo de almacenamiento en la red (SAN) que no es más que el nodo donde se encuentran almacenados los materiales audiovisuales.

2.7 Estándar de codificación

El estándar de codificación es muy importante en cualquier producto, usar técnicas de codificación sólidas y realizar buenas prácticas de programación con vistas a generar un código de alta calidad es de gran importancia para la calidad del software y para obtener un buen rendimiento. Además, si se aplica de forma continuada un estándar de codificación bien definido, se utilizan técnicas de programación apropiadas, posteriormente, se efectúan revisiones del código de rutinas, caben muchas posibilidades de que un proyecto de software sea fácil de comprender y de mantener.

Los nombres de cada uno de los elementos del programa deben ser significativos; su nombre debe explicar en lo posible el uso del elemento. Se deben declarar las variables en líneas separadas y añadir comentarios descriptivos junto a cada declaración de variables, si es necesario. La mayoría de los elementos se deben nombrar usando sustantivos, los atributos deben comenzar con letra minúscula y los métodos deben comenzar con letra mayúscula.

El nombre de las constantes debe ser un sustantivo en mayúsculas. Para separar palabras se usará el guión bajo: _.

```
function Agregar_Solicitud_Externa($tiempo_ini, $tiempo_fin, $formato, $am_id, $id_naturaleza,
    $id_usuario, $empresa, $nombre, $apellido, $ci) {
    $modelo = new recuperacion_modelo();
    $procesamiento = new Procesamiento_controlador();
    // Se declara las variables relacionadas a codificar
    $flujo = $procesamiento->Codificar_AM();
    if ($flujo == false)
        return $modelo->DB_Agregar_Solicitud_Externa($tiempo_ini, $tiempo_fin, $formato,
            $am_id, $id_naturaleza, $id_usuario, $empresa, $nombre, $apellido, $ci, "pendiente");
    return $modelo->DB_Agregar_Solicitud_Externa($tiempo_ini, $tiempo_fin, $formato,
        $am_id, $id_naturaleza, $id_usuario, $empresa, $nombre, $apellido, $ci, $flujo);
}
```

FIGURA 50: EJEMPLO DE CÓDIGO

El uso de un estándar de codificación proporcionó una mejor estructura y comprensión del código. Dicho estándar permite además una mejor organización garantizando que en caso de prescindir de un desarrollador, el que ocupe su lugar sea capaz de seguir el trabajo logrando un producto de calidad.

2.8 Conclusiones parciales

El uso de patrones permitió elaborar diagramas de clases de diseño fiables para la implementación de los diferentes casos de uso del módulo web de consulta y recuperación de materiales audiovisuales. La elaboración de los diagramas de clases del diseño, junto a otros artefactos como el modelo de datos, el diagrama de despliegue, el modelo de implementación y el estándar de codificación, sirvió de guía para la implementación del módulo y servirá de base para que otros desarrolladores mejoren sus funcionalidades y le den mantenimiento al software.

CAPÍTULO 3: PRUEBA

Este capítulo trata los elementos relacionados con la validación de la solución. Se describen los casos de prueba del sistema y se realizan las pruebas de caja negra sobre las interfaces, puntualizando los resultados obtenidos en las mismas.

3.1 Pruebas

El proceso de pruebas de un software es un instrumento adecuado para determinar el status de la calidad de un producto de software. En este proceso se ejecutan pruebas dirigidas a componentes del software o al sistema de software en su totalidad, con el objetivo de medir el grado en que el software cumple con los requerimientos. (38)

Las pruebas son una actividad en la cual el sistema se ejecuta para verificar su funcionamiento, los resultados se observan y registran para realizar una evolución del software. Las pruebas facilitan identificar los fallos que puedan existir en la implementación. Existen varios tipos de pruebas que permiten verificar y relevar la calidad de un producto de software, como son las pruebas de caja negra que se encargan principalmente en validar los datos de entrada así como la salida que se produce para cada caso.

Las pruebas de caja negra se apoyan y basan en la especificación de requisitos y documentación funcional. Se limitan a probar con datos de entrada y estudia como salen, sin preocuparse de lo que ocurre en el interior. (39)

Las pruebas de caja negra permiten encontrar

- ✚ Funciones incorrectas o ausentes.
- ✚ Errores de interfaz.
- ✚ Errores en estructuras de datos o en accesos a las Bases de Datos externas.
- ✚ Errores de rendimiento.
- ✚ Errores de inicialización y terminación.

Los casos de prueba de la caja negra pretenden demostrar que:

- ✚ Las funciones del software son operativas

- ✚ La entrada se acepta de forma correcta
- ✚ Se produce una salida correcta
- ✚ La integridad de la información externa se mantiene

Para desarrollar la prueba de caja negra existen varias técnicas, entre ellas están:

- ✚ Técnica de la Partición de Equivalencia: esta técnica divide el campo de entrada en clases de datos que tienden a ejercitar determinadas funciones del software.
- ✚ Técnica del Análisis de Valores Límites: esta técnica prueba la habilidad del programa para manejar datos que se encuentran en los límites aceptables.
- ✚ Técnica de Grafos de Causa-Efecto: es una técnica que permite al encargado de la prueba validar complejos conjuntos de acciones y condiciones. (39)

Al módulo de consulta y recuperación de materiales audiovisuales se le realizaron pruebas de caja negra pues estas se centran en lo que se espera de la aplicación, es decir, intentan encontrar casos en los que el módulo no funcione de manera correcta. Las pruebas de caja negra están especialmente indicadas en aquellos módulos que van a presentar una interfaz para interactuar con el usuario.

La técnica empleada para realizar esta prueba fue la técnica de partición de equivalencia que es una de las más efectivas pues permite examinar los valores válidos e inválidos de las entradas existentes en el software. La partición equivalente se dirige a la definición de casos de pruebas que descubran clases de errores, reduciendo así en número de clases de prueba que hay que desarrollar.

3.2 Casos de Prueba

La realización de las pruebas de caja negra al módulo para la consulta y recuperación de materiales audiovisuales para el proyecto Publicación y catalogación de Medias, requirió el diseño de los casos de prueba en correspondencia con las descripciones de los casos de uso. Los casos de prueba permiten comprobar todos los flujos de información del sistema y validar que este cumple con los requisitos del cliente. A continuación se presentan los casos de prueba utilizados para los casos de uso más importantes del módulo:

3.2.1 Diseño de caso de prueba para el caso de uso Gestionar Solicitud

Caso de prueba					
Nombre de la sección	Escenarios de la sección	Descripción de la funcionalidad	Flujo central	Resultados Esperados	Resultados Obtenidos
SC 1: Gestionar Solicitud.	EC 1.1: Crear la solicitud correctamente.	El usuario selecciona el material que desea solicitar, introduce los datos correspondientes y el sistema lo agrega a las solicitudes del usuario	Módulo Recuperación/Luego de obtener el material deseado, introduces los datos/Botón Guardar	El sistema debe mostrar un mensaje que indique que se añadió correctamente la solicitud y añadirse a la lista de solicitudes.	El sistema muestra la solicitud realizada en la lista de solicitudes del usuario.
	EC 1.2: No se puede crear la solicitud correctamente	El usuario selecciona el material que desea solicitar, pero no llena los datos de la solicitud y al presionar el botón Guardar, se muestra un mensaje de error para que complete todos los datos.	Módulo Recuperación/Luego de obtener el material deseado, introduces los datos/Botón Guardar	El sistema debe mostrar un mensaje indicando que debe completar todos los datos.	El sistema muestra el mensaje esperado.
	EC 1.3: Eliminar	El usuario elimina una solicitud creada (mientras su estado	Módulo Recuperación/Lista de	El sistema debe quitar de la lista de	El sistema quita de la lista la

Solicitud.	sea en espera o codificando), de la lista y el sistema elimina sus datos.	Solicitudes Solicitudes/Botón Eliminar.	solicitudes creadas la que fue eliminada por el usuario y mostrar un mensaje que indique que la solicitud fue eliminada.	solicitud eliminada y muestra el mensaje esperado.
EC 1.4: Modificar Solicitud	El usuario selecciona la solicitud que desea modificar y aparece de nuevo la solicitud con los datos para que modifique los que desee.	Módulo Recuperación/Lista de Solicitudes Solicitudes/Botón Modificar.	El sistema muestra la interfaz para modificar los datos de la solicitud y luego de modificarla debe mostrar un mensaje diciendo que fue modificada correctamente.	El sistema muestra la interfaz para modificar la solicitud con los datos correspondientes a esa solicitud y muestra el mensaje esperado.

TABLA 1: CASO DE PRUEBA PARA EL CASO DE USO GESTIONAR SOLICITUD.

3.2.2 Caso de prueba para el caso de uso Atender Solicitud

Caso de prueba					
Nombre de la sección	Escenarios de la sección	Descripción de la funcionalidad	Flujo central	Resultados Esperados	Resultados Obtenidos
SC 1: Atender Solicitud	EC 1.1: Se pueden obtener las solicitudes creadas saber su estado y descargar el material seleccionado.	El usuario busca la solicitud que desea para conocer su estado o descargar la media.	Módulo Recuperación Lista de Solicitudes	El sistema debe mostrar la tabla de resultados con las solicitudes existentes que cumplan con los filtros de búsqueda seleccionados	El sistema muestra la tabla de resultados correspondientes a la búsqueda.

	EC 1.2: No se puede atender la solicitud	El usuario busca la solicitud que desea para conocer su estado o descargar la media.	Módulo Recuperación Lista de Solicitudes	El sistema debe mostrar un mensaje indicando que no puede comunicarse con el gestor de procesos de media por lo que no puede codificar el material.	El sistema muestra el mensaje indicando que no puedo establecerse la comunicación con el gestor de procesos de media.

TABLA 2: CASO DE PRUEBA PARA EL CASO DE USO ATENDER SOLICITUD

3.2.3 Caso de prueba para el caso de uso Descargar Solicitud

Caso de prueba					
Nombre de la sección	Escenarios de la sección	Descripción de la funcionalidad	Flujo central	Resultados Esperados	Resultados Obtenidos
SC 1: Descargar Material.	EC 1.1: El usuario selecciona la opción de descargar el material	La solicitud fue atendida y su estado es de listo, por lo que el usuario podrá descargar el material.	Módulo Recuperación/Lista de Solicitudes/Botón Descargar Material	El sistema debe permitir descargar el material que desea el usuario y muestra una interfaz permitiéndole al usuario seleccionar el lugar de la descarga.	El sistema permite descargar el material que desea el usuario.
	EC 1.2: El usuario no puede	El usuario quiere descargar el material aun	Módulo Recuperación/Lista de	El sistema no muestra el botón que	El sistema no permite descargar el

	descargar el material seleccionado	cuando su estado no es listo.	Solicitudes/Botón Descargar Material	permite realizar la descarga.	material.
--	------------------------------------	-------------------------------	--------------------------------------	-------------------------------	-----------

TABLA 3: CASO DE PRUEBA PARA EL CASO DE USO DESCARGAR SOLICITUD

3.3 Resultados

Las pruebas de caja negra fueron realizadas de manera precisa a partir de los casos de prueba elaborados. En la culminación de este proceso se obtuvo como resultado que, exceptuando la comunicación con el gestor de procesos de media, las funcionalidades del módulo responden como se definió al inicio del proyecto. La no conformidad con respecto a la comunicación con el gestor fue resuelta posterior a su hallazgo y actualmente ya se comunica permitiendo convertir la media al formato seleccionado, y la posterior descarga del material. Para cada uno de estos se tuvo en cuenta las diferentes situaciones que podían existir con el fin de evaluar el comportamiento de la funcionalidad en cada situación, obteniéndose como resultado la igualdad con la especificación de los requisitos, lo que demuestra la correcta implementación de los mismos.

Una vez que se realizaron las pruebas de estrés estas arrojaron como resultado que el módulo anterior tuvo un tiempo de respuesta mayor al tiempo de respuesta del nuevo módulo.

3.4 Conclusiones Parciales

La realización de este capítulo permitió validar los requisitos identificados mediante las pruebas funcionales para el módulo consulta y recuperación de materiales audiovisuales, obteniéndose resultados satisfactorios que validan la solución del problema presentado. La realización de pruebas de caja negra permitió detectar, documentar y solucionar los errores existentes en el sistema implementado. Además concluir el período de pruebas permitió obtener una aplicación que responde correctamente a las peticiones del usuario, cumpliendo con los objetivos planteados para el módulo web de consulta y recuperación de materiales audiovisuales del proyecto Catalogación y publicación de medias.

CONCLUSIONES GENERALES

En el diagnóstico del problema fue detectada la necesidad de gestionar las solicitudes realizadas por un usuario y de descargar el material cuando este se haya convertido al formato deseado. A partir de la caracterización de los procesos relacionados con la consulta y recuperación de materiales audiovisuales, fue posible comprender e implementar con mayor precisión las funcionalidades correspondientes. El trabajo de ingeniería realizado durante el desarrollo de la aplicación posibilitó documentar detalladamente los artefactos generados en los flujos de trabajo de la metodología seleccionada, con el objetivo de que los mismos sean utilizables en versiones futuras. Además se obtuvo como producto principal de este trabajo el módulo web para la consulta y recuperación de materiales audiovisuales para el proyecto Catalogación y publicación de Medias. Dicho producto cumple con las funcionalidades establecidas, mejorando el acceso a las medias en las instituciones donde sea desplegado. El uso de tecnologías y herramientas libres para el desarrollo del módulo permitió cumplir con la política de soberanía tecnológica por la que aboga el país.

RECOMENDACIONES

Al concluir el presente trabajo, se recomienda:

- ✚ Realizar un estudio del proceso de cómo permitir al usuario gestionar la manera de visualizar el resultado de la búsqueda, ya sea agrupado por tipologías o todos los existentes.
- ✚ Formalizar el estudio de los procesos que se pueden efectuar cuando los usuarios realicen una solicitud para que los mismos tengan la posibilidad de escoger varios materiales audiovisuales.

TRABAJOS CITADOS

- 1- **Soler, Ana Paula.** *Slideshare.* [En línea] 3 de Mayo de 2011. <http://www.slideshare.net/ana2pau/materiales-audiovisuales>.
- 2- **UNESCO.** [En línea] [Citado el: 8 de Noviembre de 2012.] www.unesco.org/webworld/publications/philos/philos3.html
- 3- **Wordreference.com.** [En línea] [Citado el: 6 de Noviembre de 2012.] <http://www.wordreference.com>.
- 4- **BÚSQUEDA Y RECUPERACIÓN DE INFORMACIÓN.** [En línea] [Citado el: 6 de Noviembre de 2012.] http://www.mariapinto.es/e-coms/recu_infor.htm.
- 5- **Visualizacion.info.** [En línea] [Citado el: 6 de Noviembre de 2012.] <http://www.visualizar.info/glosario/visualizacion-de-informacion/>.
- 6- **Real Academia Española.** [En línea] [Citado el: 8 de Noviembre de 2012.]
- 7- **Masadelante.com.** [En línea] [Citado el: 8 de Noviembre de 2012.] <http://www.masadelante.com/faqs/download>.
- 8- **Gabriel H. Tolosa, Fernando R.A. Bordignon.** *Introducción a la Recuperación de Información.* Argentina: s.n., 2007.
- 9- **Gizmos Freeware Reviews.** [En línea] 15 de Octubre de 2012. [Citado el: 3 de diciembre de 2012.] <http://www.techsupportalert.com/top-5-video-streaming-websites.htm>.
- 10- **Itaso. Un mar de cosas.** [En línea] [Citado el: 3 de Diciembre de 2012.] <http://www.itsaso.com/>.
- 11- **Hernando, Roberto.** *Metodología de desarrollo de software.* [En línea] 2009. [Citado el: 24 de Enero de 2013.] http://www2.rhernando.net/modules/tutorials/doc/ing/met_soft.html.
- 12- **Pressman, Roger Putnam y Myers.** *Five core metrics.* s.l.: Dorset House, 2003.
- 13- **Pérez, Fabián Bermeo.** *Metodología RUP - desarrollo de software de calidad.* [En línea] 2010. [Citado el: 24 de Enero de 2013.] <http://fabianbermeop.blogspot.com/2010/12/metodologia-rup-desarrollo-de-software.html>.
- 14- **Alonso, Evelyn.** *Herramientas CASE para el proceso de desarrollo de Software.* [En línea] [Citado el: 16 de enero de 2012.] www.monografias.com.
- 15- **Sierra, Daniel.** *Visual Paradigm For Uml.* [En línea] 15 de Noviembre de 2007. [Citado el: 24 de Enero de 2013.] <http://www.slideshare.net/vanquishdarkenigma/visual-paradigm-for-uml>.
- 16- **Universidad de las Américas Puebla.** *Colección de Tesis Digitales.* [Citado el: 15 de 1 de 2013.]
- 17- **Kioskea.net.** [En línea] Diciembre de 2012. <http://es.kioskea.net/contents/langages/langages.php3>.

- 18- **Instituto Tecnológico de Veracruz.** [En línea] [Citado el: 1 de febrero de 2013.] <http://www.prograweb.com.mx/pweb/0202ladoCliente.html>.
- 19- **Hooping.net.** *Características y elementos básicos del lenguaje html para el diseño web.html.* [En línea] 2008. [Citado el: 23 de Enero de 2013.]
- 20- **Carrión, Nuria Rubio.** *Conocimientos básicos de Javascript.* [En línea] 2006. [Citado el: 23 de Enero de 2013.] www.Javascript.html#inicio.
- 21- **Blog OpenAlfa.** [En línea] 2012. <http://blog.openalfa.com/introduccion-a-jquery/>.
- 22- **Monografias.com.** [En línea] 25 de Noviembre de 2012. [www.monografias.com/Caracteristicas del PHP...html](http://www.monografias.com/Caracteristicas-del-PHP...html).
- 23- **Javier J. Gutiérrez.** . *CSSBlog.es.* CSSBlog.es. [Citado el: 4 de 2 de 2013.] <http://www.cssblog.es/guias/Framework.pdf>.
- 24- **D, Pablo D Torres.** SlideShare Inc. [En línea] 15 de Febrero de 2011. <http://www.ajpdsoft.com/modules.php?name=Encyclopedia&op=content&tid=795>
- 25- **Buenastareas.com.** [En línea] Marzo de 2012. [Citado el: 4 de Febrero de 2013.] <http://www.buenastareas.com/ensayos/Lenguajes-Del-Lado-Del-Servidor/3642799.html>.
- 26- *Drupal7 disponible en todo el mundo* [En línea]. [Citado el:1 de febrero de 2013.] <http://drupal.org/press/drupal-7.0/es>
- 27- **MixProgramas.com.** [En línea] 2012. [Citado el: 1 de febrero de 2013.] <http://www.mixprogramas.com/netbeans-ide-7-0-beta-2/>.
- 28- **Álvarez, Sara.** *desarrolloweb.com.* *Sistemas gestores de bases de datos.* [En línea] 31 de Julio de 2007. [Citado el: 5 de Febrero de 2013.] <http://www.desarrolloweb.com/articulos/sistemas-gestores-bases-datos.html>.
- 29- **Pecos, Daniel.** *PostGreSQL vs. MySQL.* [En línea] [Citado el: 5 de Febrero de 2013.] http://danielpecos.com/docs/mysql_postgres/x15.html.
- 30- **Salazar, Cristian.** SlidesHare. [En línea] 28 de Septiembre de 2012. [Citado el: 8 de Abril de 2013.] <http://www.slideshare.net/csalazarc/modelo-de-datos-14506949>

- 31- **L. Bass, P. Clements, R. Kazman**, *Software Architecture in Practice*, 2nd Edition, Addison Wesley, 2003
- 32- "*Estilos De Arquitectura De Software*" BuenasTareas.com. 01 2013. 02 2013
<<http://www.buenastareas.com/ensayos/Estilos-De-Arquitectura-De-Software/4999131.html>>.
- 33- **Tedeschi, Nicolás**. *MSDN*. [En línea] 2013. [Citado el: 28 de Marzo de 2013.]
<http://msdn.microsoft.com/es-es/library/bb972240.aspx>.
- 34- **Mora, Roberto Canales**. *Adictos Al Trabajo*. [En línea] 22 de Diciembre de 2003. [Citado el: 9 de Abril de 2013.]
- 35- **Larman, Prentice Hall** *UML y Patrones. Introducción al análisis y diseño orientado a objetos*
- 36- **Figuerola, Pablo**. *webdocs*. [En línea] [Citado el: 10 de Abril de 2013.]
<http://webdocs.cs.ualberta.ca/~pfiguero/soo/uml/implementacion01.html>.
- 37- **Sparx Systems**. [En línea] [Citado el: 10 de abril de 2013.]
http://www.sparxsystems.com.ar/resources/tutorial/uml2_deploymentdiagram.html.
- 38- *Pruebas de Software. Gestión de Calidad y pruebas de software*. [En línea] 2005. [Citado el: 29 de Abril de 2013.] <http://pruebasdesoftware.com/laspruebasdesoftware.htm>.
- 39- **Adán, Víctor Gómez**. *Globe Testing*. [En línea] 14 de Agosto de 2012. [Citado el: 30 de Abril de 2013.] <http://www.globetesting.com/2012/08/pruebas-de-caja-negra/>.
- 40- **Ampuero Felipe** [En línea] 18 de febrero de 2013. [Citado el: 19 de mayo de 2013.]
<http://www.googleek.info>
- 41 - **James Thornton**. *Softonic. Download videos, photos and fun animations galore* [En línea] 16 de marzo 2012. [Citado el: 19 de mayo de 2013.]
- 42- [En línea] 18 de febrero de 2013. [Citado el: 19 de mayo de 2013.] <http://www.webtvhub.com/>

BIBLIOGRAFÍA

- Altamirano, Ing. Alfonso Valdez. 2009.** Comparativo de Entornos de Desarrollo Integrados (IDE's). 2009.
- Alonso, Evelyn.** Monografias.com. Herramientas CASE para el proceso de desarrollo de Software. [En línea] [Citado el: 16 de enero de 2012.] www.monografias.com.
- BÚSQUEDA Y RECUPERACIÓN DE INFORMACIÓN.** [En línea] [Citado el: 6 de Noviembre de 2012.] http://www.mariapinto.es/e-coms/recu_infor.htm
- Camacho, Erika, Cardeso, Fabio y Núñez, Gabriel. 2004.** Arquitecturas de software. 2004.
- Carrión, Nuria Rubio. Conocimientos básicos de Javascript.** [En línea] 2006. [Citado el: 23 de Enero de 2013.] www.Javascript.html#inicio.
- Denzer, Patricio. 2002.** PostgreSQL. 2002.
- DRAE. 2001.** REAL ACADEMIA ESPAÑOLA. REAL ACADEMIA ESPAÑOLA. [En línea] 2001. [Citado el: 15 de 11 de 2012.] <http://rae.es/rae.html>.
- Gabriel H. Tolosa, Fernando R.A. Bordignon.** Introducción a la Recuperación de Información. Argentina: s.n., 2007
- Pecos, Daniel.** PostGreSQL vs. MySQL. [En línea] [Citado el: 5 de Febrero de 2013.] http://danielpecos.com/docs/mysql_postgres/x15.html.
- Pressman, Roger S. 2010.** Ingeniería de Software: un enfoque práctico. 2010.
- Pruebas de Software. Gestión de Calidad y pruebas de software.** [En línea] 2005. <http://pruebasdesoftware.com/laspruebasdesoftware.htm>.
- Sánchez, María A. Mendoza. 2004.** Metodologías De Desarrollo De Software. **Metodologías De Desarrollo DeSoftware.** [En línea] 2004.[Citadoel:8de1de 2013.] http://www.informatizate.net/articulos/metodologias_de_desarrollo_de_software_07062004.html
- UNESCO.** [En línea] [Citado el: 8 de Noviembre de 2012.] www.unesco.org/webworld/publications/philos/philos3.html
- UML y Patrones. Introducción al análisis y diseño orientado a objetos - Larman - Prentice Hall**

ANEXOS

1- Requisitos Funcionales

1.1. Realizar búsquedas de material audiovisual.

1.1.1. Realizar búsqueda por clasificación.

Descripción: El sistema permitirá la realización de búsqueda de videos que están almacenadas en el servidor. Se permitirá establecer filtros referentes a la clasificación del material y a los atributos específicos de la clasificación.

1.1.2. Realizar búsqueda avanzada.

Descripción: El sistema permitirá realizar búsquedas definiendo filtros avanzados relativos a la fecha o a la descripción del material, lo que posibilitará la búsqueda más específica de materiales atendiendo a las fechas almacenadas y a las descripciones asociadas a los mismos.

1.1.3. Realizar búsqueda básica.

Descripción: El sistema permitirá la realización de búsqueda de videos que están almacenados en el servidor. Se permitirá establecer filtros básicos en correspondencia con los datos que han sido asociados en la catalogación de materiales audiovisuales digitalizados.

1.1.4. Realizar búsqueda de materiales a catalogar.

Descripción: El sistema permitirá la realización de búsqueda de videos que están almacenados en el servidor que no hayan sido catalogados.

1.1.5. Limpiar filtros de búsqueda.

Descripción: El sistema deberá permitir limpiar todos los filtros de búsqueda insertados por el usuario con el objetivo de introducir nuevos para obtener otros resultados.

1.2. Reproducir material audiovisual.

1.2.1. Reproducir media.

Descripción: El sistema permitirá reproducir el streaming de video en baja resolución asociado a los materiales almacenados en el servidor. Se debe permitir realizar acciones de reproducción con el fin de facilitar la catalogación audiovisual y la consulta.

1.2.2. Comenzar Reproducción.

Descripción: El sistema debe permitir comenzar la reproducción de un video.

1.2.3. Pausar Reproducción.

Descripción: El sistema debe permitir pausar la reproducción de la media que se está reproduciendo.

1.2.4. Parar Reproducción.

Descripción: El sistema debe permitir parar la reproducción del material en curso. En este caso se regresa al tiempo de inicio de reproducción del video.

1.2.5. Retrasar Reproducción.

Descripción: El sistema debe permitir regresar la reproducción del video la cantidad de segundos que ha especificado el usuario.

1.2.6. Adelantar Reproducción.

Descripción: El sistema debe permitir adelantar la reproducción del video la cantidad de segundos que ha especificado el usuario.

1.2.7. Ir a un Instante de Tiempo.

Descripción: El sistema debe permitir el posicionamiento directo de la reproducción a un instante de tiempo optado por el usuario sobre la línea de reproducción del material.

1.2.8. Controlar Volumen.

Descripción: El sistema permitirá aumentar o disminuir el volumen de la reproducción.

1.2.9. Cambiar Velocidad de Reproducción.

Descripción: El sistema de permitir cambiar la velocidad de reproducción del video.

1.2.10. Crear Marcas.

Descripción: El sistema debe permitir establecer marcas de inicio y fin en algún tiempo del video en reproducción.

1.2.11. Desplazarse hacia Marca Inicio.

Descripción: El sistema debe brindar la posibilidad de desplazar el cursor de reproducción hasta la marca de inicio previamente ubicada.

1.2.12. Desplazarse hacia Marca Fin.

Descripción: El sistema debe brindar la posibilidad de desplazar el cursor de reproducción hasta la marca de fin previamente ubicada.

1.2.13. Reproducir Fragmento entre Marcas.

Descripción: El sistema debe permitir reproducir un fragmento de video definido por las marcas de inicio y fin.

1.2.14. Ir a un Instante de Tiempo Preciso.

Descripción: El sistema debe permitir el posicionamiento directo de la reproducción en un instante de tiempo definido numéricamente por el usuario.

1.2.15. Reproducir a Pantalla Completa.

Descripción: El reproductor deberá permitir visualizar la reproducción en la pantalla completa.

1.3. Gestionar Solicitud.

1.3.1. Insertar Solicitud

Descripción: El sistema debe permitir realizar la solicitud de material audiovisual indicando sus datos de identidad y los datos del material que desea.

1.3.2. Modificar Solicitud

Descripción: El sistema debe permitir a los usuarios modificar los datos de solicitudes de material audiovisual que han realizado.

1.3.3. Eliminar Solicitud

Descripción: El sistema debe permitir a los usuarios externos o internos del canal eliminar los datos de la solicitud de material audiovisual que ha realizado.

1.3.4. Atender Solicitud

1.3.4.1. Buscar Solicitud

Descripción: El sistema debe permitir acceder a los datos de las solicitudes realizadas por los usuarios para darle atención a las mismas.

1.3.4.2. Responder Solicitud

Descripción: El sistema debe permitir responder a las solicitudes que han sido realizadas por los usuarios.

1.3.5. Descargar Material

Descripción: El sistema debe permitir a los usuarios con permisos descargar los materiales que han sido solicitados.

2-Diagrama de los casos de uso del sistema

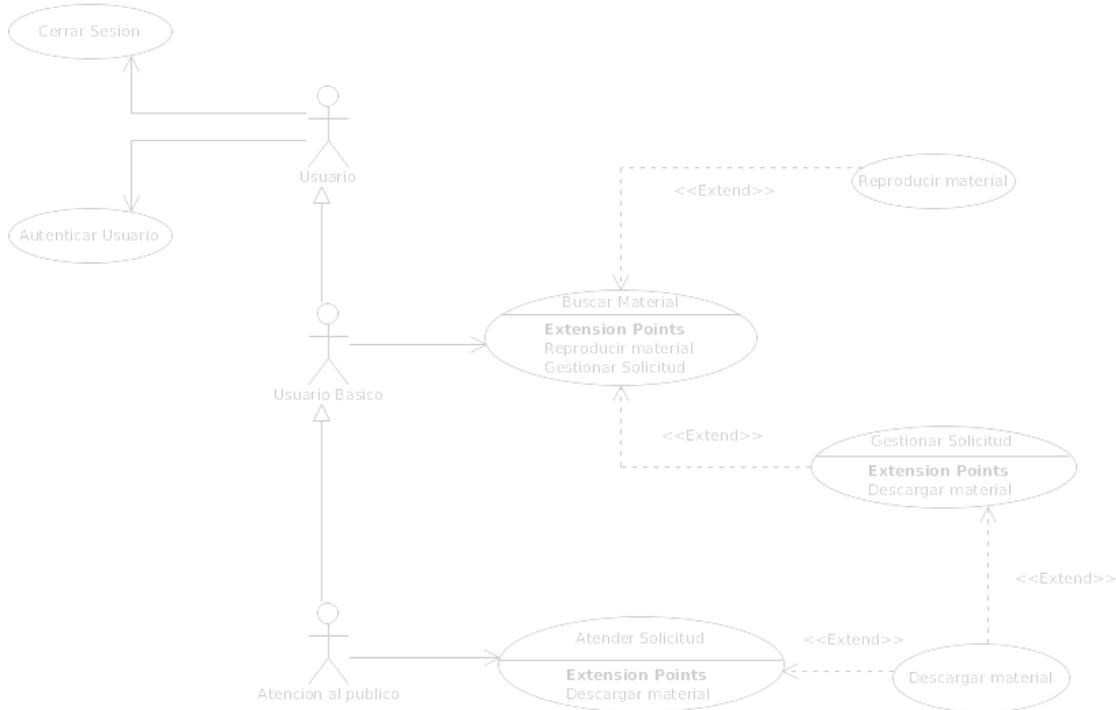


FIGURA 11: DIAGRAMA DE CASOS DE USO DEL SISTEMA

3- Descripción de los casos de uso

3.1- Gestionar Solicitud

Objetivo	Realizar, modificar o eliminar la solicitud de las medias deseadas en el sistema.
Actores	Usuario básico: (Inicia) Inserta, edita y elimina los datos de una solicitud.
Resumen	El caso de uso inicia cuando el usuario selecciona adicionar, modificar o eliminar una solicitud, para ello introduce los datos necesarios, el caso de uso termina cuando se inserte, modifique o elimine una solicitud.
Complejidad	Media
Prioridad	Alta
Precondicione s	El usuario esta autenticado y tiene permisos para gestionar las solicitudes.
Poscondicione s	Se adicionó una solicitud al sistema. Se modificaron los datos de una solicitud.

	Se eliminó una solicitud.	
Flujo de eventos		
Flujo básico Gestionar solicitud		
	Actor	Sistema
1.	Selecciona una de las opciones siguientes: <ul style="list-style-type: none"> - Adicionar Solicitud. - Modificar Solicitud. Ver sección1 “Modificar Solicitud”. - Eliminar Solicitud. Ver sección2 “Eliminar Solicitud”. 	
2.		Muestra una interfaz con los datos necesarios para realizar la solicitud.
3.	Inserta los datos que se solicitan y presiona la opción Aceptar(K).(Figura12a) <ul style="list-style-type: none"> - Nombre(A) - Apellidos(B) - Identificación(C) - Institución a la que pertenece.(D) - Tiempo de inicio.(E) - Tiempo de fin.(F) - Formato(G) 	
4.		Valida que los datos estén correctos.
5.		Inserta los datos en la base de datos.
6.		Muestra un mensaje diciendo que la solicitud fue insertada correctamente. Termina el caso de uso. (Figura12b).
Flujos alternos		
4ª Datos incorrectos		
	Actor	Sistema
4.		Muestra un mensaje de error indicando que faltan datos o hay error en los mismos. Regresa al paso3 del flujo básico. (Figura12c).

Sección1: “Modificar solicitud”		
Flujo básico Modificar Solicitud		
	Actor	Sistema
2.	Selecciona la solicitud que de sea modificar.	
3.		Muestra una interfaz con los datos de la solicitud quede sea modificar.
4.	Cambia alguno de los atributos y presión a la opción Aceptar (K). (Figura12d). <ul style="list-style-type: none"> - Nombre(A) - Apellidos(B) - Identificación(C) - Institución a la que pertenece. (D). - Tiempo de inicio.(E) - Tiempo de fin.(F) - Formato(G) 	
5.		Valida que los datos insertados estén correctos.
6.		Busca en la base de datos las solicitudes de acuerdo a su identificador y modifica los datos que fueron cambiados.
7.		Muestra un mensaje indicando que la solicitud fue modificada correctamente. Termina el caso de uso. (Figura12e).
Flujos alternos		
5aDatos Incorrectos		
	Actor	Sistema
5.		Muestra un mensaje de error indicando que faltan datos o hay error en los mismos .Regresa al paso4 del flujo básico. (Figura12c).
Sección2: “Eliminar solicitud”		
Flujo básico Eliminar solicitud		

	Actor	Sistema
4.	Selecciona la solicitud que desea eliminar. Y presiona el botón “Eliminar”. (Figura12f).	
5.		Muestra un mensaje de confirmación para saber si desea o no eliminarla solicitud.(Figura12g)
6.	Selecciona la opción “Aceptar”.	
7.		Elimina la solicitud. Termina el caso de uso.
Relaciones	CU Incluidos	No existen.
	CU Extendidos	Descargar Material en el CU Gestionar Solicitud.
Requisitos no funcionales	RNF1. RNF 2. RNF 6. RNF 8. RNF 12. RNF 12.1. RNF 12.2. RNF 12.3. RNF 13.	
Asuntos pendientes	No existen.	

[Reservaciones de la Biblioteca](#)
[Datos Solicitud](#)
[Lista de Solicitudes](#)

Externa
 Interna

Usuario:
Nombre:
Título:

Apellidos:
Tiempo inicio:

CI:
Tiempo final:

Institución:

FIGURA 6: A



FIGURA 7: B

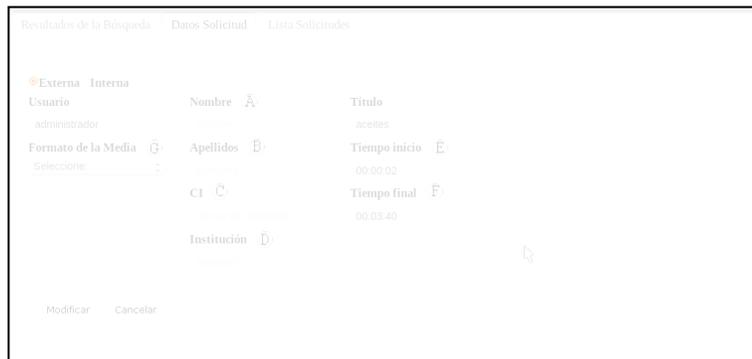


FIGURA 8: C



FIGURA 12: D

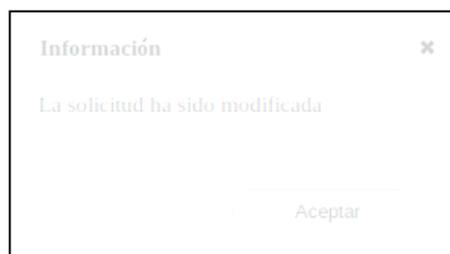


FIGURA 9: E

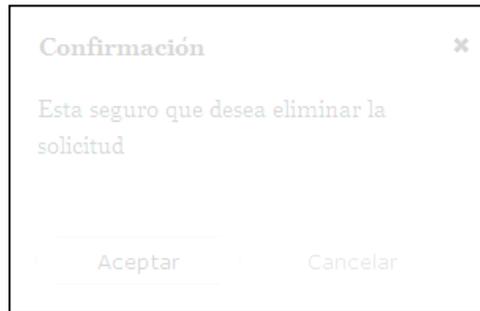


FIGURA 12: F

Titulo	Fecha	Duración	Estado	Opciones
aceites	05/14/2013	34 min	Error en la Conversión	✕ ✓
aceites	05/14/2013	34 min	Error en la Conversión	✕ ✓
aceites	05/14/2013	34 min	Error en la Conversión	✕ ✓
aceites	05/14/2013	34 min	Error en la Conversión	✕ ✓
aceites	05/14/2013	34 min	Error en la Conversión	✕ ✓

Página 1 de 3

FIGURA 12: G

3.2- Atender

Solicitud

Objetivo	Dar respuesta a las solicitudes hechas por el usuario básico.
Actores	Atención al público:(Inicia) Buscar las solicitudes hechas por el usuario básico.
Resumen	El caso de uso inicia cuando el usuario realiza una búsqueda de las solicitudes hechas para darle atención a las mismas. El caso de uso termina cuando sean atendidas las solicitudes.
Complejidad	Alta
Prioridad	Media
Precondiciones	El usuario esta autenticado y tiene permisos para atender las solicitudes.
Poscondiciones	Solicitud Atendida.
Flujo de eventos	

Flujo básico Atender Solicitud		
	Actor	Sistema
1.	<p>Introduce los datos para buscar la solicitud deseada teniendo en cuenta los siguientes parámetros:(Figura13a)</p> <ul style="list-style-type: none"> - Fecha inicio(A) - Fecha Fin(B) - Nombre(C) - Apellidos (D) - CI(E) - Estado(F) - Institución (G) - Usuario (H) 	
2.		Valida que los datos sean correctos.
3.		Muestra la lista de solicitudes encontradas de acuerdo a los parámetros establecidos, indicando el estado en que se encuentran (codificando, lista).En caso de estar listas permite la opción de descargar el material. Termina el caso de uso.(Figura13b)
Flujos alternos		
2aDatosIncorrectos		
	Actor	Sistema
2.		Muestra un mensaje de error indicando que faltan datos o hay error en los mismos. Regresa a lpaso1del flujo básico.(Figura13c).
Relaciones	CU Incluidos	No existen.
	CU Extendidos	Descargar Material Paso3 del Atender Solicitud.
Requisitos no funcionales	RNF1. RNF 2.	

	RNF 6. RNF 8. RNF 12. RNF 12.1. RNF 12.2. RNF 12.3. RNF 13.
Asuntos pendientes	No existen.

Estado	Nombre	CI	Apellidos	Institución
Lista : F	Nombre	C	E Apellidos	D Institución G
Usuario	Fecha			
Usuario	H 05/06/2013	A		
	05/14/2013	B		

FIGURA 13: A

Título	Fecha	Duración	Estado	Opciones
aceites	05/14/2013	34 min	Error en la Conversión	* ✓
aceites	05/14/2013	34 min	Error en la Conversión	* ✓
aceites	05/14/2013	34 min	Error en la Conversión	* ✓
aceites	05/14/2013	34 min	Error en la Conversión	* ✓
aceites	05/14/2013	34 min	Error en la Conversión	* ✓

Página 1 de 3

FIGURA 13: B



FIGURA 13: C

3.3- Descargar Material

Objetivo	Descargar los materiales que el usuario solicitó.	
Actores	Usuario básico, Atención al público:(Inicia) Descarga los materiales solicitados.	
Resumen	El caso de uso inicia cuando el usuario atiende o gestión a las solicitudes y desea descargar el material. El caso de uso termina cuando se descarguen las medias deseadas.	
Complejidad	Alta	
Prioridad	Media	
Precondiciones	El usuario esta autenticado y tiene permisos para descargar una media. El estado de la media debe ser Listo.	
Poscondiciones	Descarga el material.	
Flujo de eventos		
Flujo básico Descargar Material		
	Actor	Sistema
1.	Selecciona la opción "Descargar".(A)(Figura 14)	
2.		Muestra una interfaz para que el usuario indique la dirección donde desea almacenar la media.
3.	Selecciona el lugar donde se guardará la media.	
Relaciones	CU Incluidos	No existen.

	CU Extendidos	No existen.
Requisitos no funcionales	RNF1. RNF 2. RNF 6. RNF 8. RNF 12. RNF RnF12.1. RNF 12.2. RNF 12.3. RNF 13.	
Asuntos pendientes	No existen.	

Título	Fecha	Duración	Estado	Opciones
aceites	05/14/2013	34 min	Listo	* ✓ ▾ A
aceites	05/14/2013	34 min	Error en la Conversión	* ✓
aceites	05/14/2013	34 min	Error en la Conversión	* ✓
aceites	05/14/2013	34 min	Error en la Conversión	* ✓
aceites	05/14/2013	34 min	Error en la Conversión	* ✓

Página 1 de 3

FIGURA 10

4- Requisitos No Funcionales

1-Usabilidad

Tipo de usuario final.

La aplicación debe ser manejada tanto por usuarios domésticos como por empleados de instituciones con conocimientos básicos en informática.

2- Tipo de Aplicación Informática.

Aplicación Informática: WEB.

3- Finalidad

Proveer a una televisora, centro educativo, ministerio o cualquier otra institución que lo requiera de una solución tecnológica que permita gestionar, catalogar y recuperar su información audiovisual con la finalidad de preservar y conservar el patrimonio social, político, cultural y humanístico, permitiendo que la misma esté disponible para su uso en el desarrollo de nuevos materiales o se encuentre a disposición de sus usuarios mediante su publicación en la web.

4-Ambiente.

Estos requisitos están en función de la dimensión del entorno donde se despliegue, estos son los requisitos mínimos con los que deberá contar la solución que se propone:

Estaciones clientes: Memoria RAM: 1 GB, Procesador: Core 2 Duo, Monitor: 17 ".

Servidor de procesamiento (codificación e indexación): Arquitectura 64 bits (x64), Procesador: 1 QuadCore, Memoria RAM: 4 GB.

Servidor de streaming: Arquitectura 64 bits (x64), Procesador: 1 QuadCore, Memoria RAM: 8 GB.

Servidor web: Arquitectura 64 bits (x64), Procesador: 1 QuadCore, Memoria RAM: 4 GB.

Servidor de BD: Arquitectura 64 bits (x64), Procesador: 1 QuadCore, Memoria RAM: 4 GB.

5-Confiabilidad

Confiabilidad del Sistema Captura y publicación de medias.

RNF 5.1 Si se interrumpe la energía o la red en algunas de las PC clientes, estas no podrán prestar el servicio, se mostrara un mensaje indicando que no hay conexión con el servidor.

RNF 5.2 Si se interrumpe la energía o la red en algunas de la PC servidora se verán afectadas las PC clientes pues el sistema no se mostrará.

RNF 5.3 Al reanudarse la energía o la red en las PC clientes afectadas, podrán visualizarse nuevamente las funcionalidades del sistema.

RNF 5.4 Al reanudarse la energía o la red en las PC servidoras afectadas, podrán visualizarse nuevamente las funcionalidades del sistema en las PC clientes. Hay que levantar el sistema manualmente.

RNF 5.5 El sistema puede utilizar tarjetas captura, si la comunicación con dicha tarjeta falla el sistema no podrá publicar señales en vivo.

RNF 5.6 Si se le cambia la hora a las PC clientes y se tiene en cuenta dicha hora para realizar las operaciones, pueden ocurrir fallos pues todas las operaciones del sistema se realiza teniendo en cuenta la hora configurada en las PC servidores.

6-Eficiencia

Eficiencia del Sistema Captura y publicación de medias.

Tiempo de respuesta por transacción (promedio: 3, máximo: 5).

Rendimiento (500 transacciones por segundo).

Capacidad (2000 clientes).

Utilización de recursos (16 gb ram, disco 1tb, comunicaciones 1 GB de red, tarjeta de captura).

7- Soporte

RNF 7.1 Todas las **estaciones clientes** tienen que tener interfaz de red, cualquier Sistema Operativo, cualquier navegador excepto Internet Explorer, recomendado navegador Mozilla Firefox 10 o superior. Para la funcionalidad de reproducir en navegador externo, tener instalado reproductor VLC.

RNF 7.2 Servidores: En todos el SO Ubuntu 12.04, Servidor de streaming: Flumotion 0.6.1, Servidor de BD: PostgreSQL 9, Servidor web: Drupal 7, apache 2, Servidor de procesamiento: ffmpeg, Mencoder, ffmpeg2theora v2.7.

8-Restricciones de diseño

Restricciones de diseño del Sistema Captura y publicación de medias.

- **CMS:** Drupal 7.X.

- **Lenguaje:** PHP 5.3 o superior.
- **Gestor de BD:** PostgreSQL 9.1 o superior.
- **Servidor de streaming:** Flumotion 10.0 o superior.
- **Biblioteca de javascript:** jQuery 1.4.4 o superior.

Requisitos para la documentación de usuarios en línea y ayuda del sistema.

9- Ayuda de usuarios del Catalogación y publicación medias.

El sistema deberá brindar a sus clientes finales, manuales de usuarios por roles en los que se especificarán todas las funciones que pueden realizar y cómo hacerlo.

Componentes Comprados

No existen.

Interfaz

10- Interfaces de usuario del Sistema Catalogación y publicación medias.

Las interfaces gráficas implementadas por el sistema deben concebirse con un ambiente sencillo y de navegación fácil para el usuario. Los colores serán convenientemente utilizados dada la funcionalidad y objetivo del sistema, siendo claros en la mayor parte de la aplicación logrando una vista agradable a los usuarios y resaltando con otras tonalidades los mensajes de interacción de los que dependen las funcionalidades críticas.

Interfaces de Comunicación del Sistema Catalogación y publicación medias.

RNF 11.1 El sistema requiere interactuar con el componente de Administración y Seguridad para la autenticación y gestión de usuarios, permisos y roles.

RNF 11.2 El sistema requiere interactuar con el componente de Catalogación para la descripción de medias.

Requisitos de Licencia

No existe.

Requisitos Legales, de Derecho de Autor y otros.

11- Requisitos legales y de derecho de autor.

RNF 12.1 El sistema debe ajustarse y regirse por la ley, decretos leyes, decretos, resoluciones y manuales (órdenes) establecidos, que norman los procesos que serán automatizados.

RNF 12.2 Como producto, se distribuirá amparado bajo las normativas legales establecidas en el registro comercial emitido por las entidades jurídicas de la Universidad de las Ciencias Informáticas.

RNF 12.3 El producto debe ser registrado en el CENDA para proteger los derechos de autor.

Estándares Aplicables

12- Estándares aplicables al Sistema Catalogación y publicación de medias.

El desarrollo del software estará regido por las normas establecidas para el Nivel 2 de CMMI aplicado en la UCI. Serán adoptadas las políticas establecidas en Procesos y Guías 3.3\Documentos Generales de los Procesos\Políticas.pdf.

GLOSARIO

CASE: Computer Aided Software Engineering. Bajo el término de Ingeniería de Software Asistida por Ordenador se incluyen una serie de herramientas, lenguajes y técnicas de programación que permiten la generación de aplicaciones de manera semiautomática. Las herramientas CASE liberan al programador de parte de su trabajo y aumentan la calidad del programa a la vez que disminuyen sus posibles errores.

Framework: define, en términos generales un conjunto estandarizado de conceptos, prácticas y criterios para enfocar un tipo de problemática particular que sirve como referencia, para enfrentar y resolver nuevos problemas de índole similar.

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

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

Píxeles: Son aquellos puntos que forman las imágenes digitales, como por ejemplo, las de la pantalla del computador, o de la imagen que resulta de una impresora. Cada uno de estos puntos se denomina píxel, y al observarlos todos juntos se forma una imagen.

Plugins: programa o aplicación que añade funcionalidad al programa principal donde está hospedado, desempeña una tarea muy específica dentro del conjunto de todas funciones del programa anfitrión.

Protocolo: Conjunto de normas y procedimientos útiles para la transmisión de datos, conocido por el emisor y el receptor

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

Streaming: Es un término que hace referencia al hecho de escuchar música o ver vídeos sin necesidad de descargarlos, sino que se hace por fragmentos enviados secuencialmente a través de la red

TCP/IP: Son las siglas de Protocolo de Control de Transmisión/Protocolo de Internet (en inglés Transmission Control Protocol/Internet Protocol), un sistema de protocolos que hacen posibles servicios Telnet, FTP, E-mail, y otros entre ordenadores que no pertenecen a la misma red.

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.

XML-RPC: es un protocolo de llamada a procedimiento remoto que usa XML para codificar los datos y HTTP como protocolo de transmisión de mensajes