

UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS
Facultad 6



**SUBSISTEMA DE CONTROL DE INGESTA DEL SISTEMA
DE CAPTURA Y CATALOGACIÓN DE MEDIAS.**

TRABAJO DE DIPLOMA PARA OPTAR POR EL TÍTULO DE INGENIERO EN
INFORMÁTICA

AUTOR: Roberto Frometa Lescay

TUTOR: Ing. Anibal Santos Santos

Ciudad de La Habana, junio, 2011.

Año 53 de la Revolución.

La Revolución no se lleva en la boca para vivir de ella, se lleva en el corazón para morir por ella.



DEDICATORIA

*Dedicado a mi linda y hermosa madre, mi eterna guía y
consejera.*

A mi linda familia...a mi padre.

Al esposo de mi madre por su apoyo incondicional.

A mis abuelos y mis hermanas.

AGRADECIMIENTOS

A mi madre por creer siempre en mí, por ser la principal fuente de inspiración en la vida. Por ser mi primera maestra y hacer de mí la persona que soy, por enseñarme a vivir, a resistir, por sus consejos y la infinita confianza que siempre ha depositado en mí, por el amor y el cariño que siempre me ha dado, porque todo lo que soy se lo debo a ella y porque se merece más de lo que puede ser capaz de dar.

A la hermosa familia que me dio la vida, a mi padre.

A todas las personas que de una forma u otra han contribuido a mi formación profesional.

A Anibal por ser un tutor excelente y por la ayuda que me ha brindado.

A los amigos que de una forma u otra me han ayudado, a Fernando, Luis

Ángel, Forge, Maylín, Rocny, Yenia, Yelen, Magalis, Dennis,

Osmany.

Muchas Gracias.

DECLARACIÓN DE AUTORÍA

Declaro que soy el único autor de este trabajo titulado “Subsistema de Control de Ingesta del Sistema de Captura y Catalogación de Medias” y autorizo a la Universidad de las Ciencias Informáticas a hacer uso del mismo en su beneficio.

Para que así conste firmo la presente a los ____ días del mes de _____ del año 2010.

Roberto Frometa Lescay

Ing. Anibal Santos Santos

DATOS DE CONTACTO

Tutor: Ing. Anibal Santos Santos

E-mail: assantos@uci.cu

- Ingeniero en Ciencias Informáticas, Universidad de las Ciencias Informáticas, 2009.
- Título de oro.

RESUMEN

El Instituto Cubano de Radio y Televisión (ICRT) es una de las instituciones más importantes de nuestro país cuyo trabajo está relacionado con archivos audiovisuales. A pesar del acelerado desarrollo tecnológico, aun existe información audiovisual almacenada en contenedores de formato analógico. En aras de aprovechar las bondades que brindan los avances tecnológicos actuales en cuanto a la preservación de la calidad de la imagen y el sonido, así como la prolongación de su vida útil, se hace imprescindible la digitalización de estos archivos. El proyecto Sistema de Captura y Catalogación de Medias (SCCM), perteneciente al departamento de Señales Digitales del centro de desarrollo GEYSED, encamina sus esfuerzos al desarrollo de un sistema de gestión de audio, video y streaming como solución para una mejor organización, catalogación, gestión y recuperación de información audiovisual. Como parte de este trabajo, se decidió llevar a cabo una investigación cuyo principal resultado sería desarrollar una aplicación Web que permita la ingesta de los nuevos materiales audiovisuales al sistema antes mencionado, esta aplicación ha sido denominada como Subsistema de Control de Ingesta. En el presente trabajo de diploma se exponen los pasos seguidos durante el desarrollo de dicho subsistema, el cual es actualmente una realidad, y que además de permitir la ingesta de las nuevas medias, proporciona considerables mejoras en cuanto al control y la organización del proceso de gestión de fallas originadas durante la digitalización de materiales audiovisuales en el ICRT. Esta aplicación constituye la primera puerta de entrada del sistema desarrollado por el equipo de trabajo del proyecto SCCM y a partir de esta se hace posible la realización de los procesos relacionados con dichos materiales tales como la catalogación, búsqueda y recuperación. La implementación estuvo basada en el uso de herramientas de software libre.

PALABRAS CLAVES

Analógico, audiovisuales, catalogación, digitalización, ingesta, streaming, subsistema.

ÍNDICE DE IMÁGENES

Imagen 1: Interfaz de Td Ficus.....	10
Imagen 2: Videoma Archivo.....	12
Imagen 3: Fases de RUP.....	15
Imagen 4: Compatibilidad con Entornos de desarrollo y lenguajes.....	17
Imagen 6: Diagrama de Casos de Uso del Negocio.	27
Imagen 7: Diagrama de Actividades del Caso de Uso del Negocio Digitalizar Material.	29
Imagen 8: Diagrama de Actividades del Caso de Uso del Negocio Editar Material.....	32
Imagen 9: Diagrama de Actividades del Caso de Uso del Negocio Describir Material.....	34
Imagen 10: Diagrama de Actividades del Caso de Uso del Negocio Almacenar Material.	36
Imagen 11: Modelo Vista Controlador.....	57
Imagen 13: Diagrama de Clases del Diseño: Gestionar Tarea.	60
Imagen 14: Modelo de Datos.	62
Imagen 15: Diagrama de Despliegue.	63
Imagen 16: Diagrama de Componentes.....	64

ÍNDICE DE TABLAS

Tabla 1: Actores del negocio y sus descripciones.....	27
Tabla 2: Trabajadores del negocio y sus descripciones.....	27
Tabla 3: Descripción textual del CUN Digitalizar Material.	29
Tabla 4: Descripción textual del CUN Editar Material.	31
Tabla 5. Descripción textual del CUN Describir Material.....	33
Tabla 6: Descripción textual del CUN Almacenar Material.	36
Tabla 7: Actores del Sistema y sus Descripciones.....	40
Tabla 8: Diagrama de Casos de Uso del Sistema.....	41
Tabla 15: Descripción textual del CUS Autenticar Usuario.....	43
Tabla 14: Descripción textual del CUS Adicionar Proyecto.	46
Tabla 11: Descripción textual del CUS Transferir Material.	47
Tabla 9: Descripción textual del CUS Gestionar Tarea.	50
Tabla 10: Descripción textual del CUS Realizar Tareas.....	52
Tabla 13: Descripción textual del CUS Adicionar Material Definitivo.....	53
Tabla 16: Secciones a probar en el caso de uso.	66
Tabla 17: Descripción de variables.....	66

ÍNDICE

INTRODUCCIÓN	1
CAPÍTULO 1: “Fundamentación Teórica”	6
1.1 Introducción.....	6
1.2 Conceptos asociados al dominio del problema.....	6
1.3 Proceso de gestión de fallas originadas durante la digitalización de materiales audiovisuales.	8
1.3.1 Descripción general.	8
1.3.2 Descripción actual del dominio del problema.	8
1.3.3 Situación problemática.....	9
1.4 Análisis de otras soluciones existentes.	10
1.4.1 Td Ficus.....	10
1.4.2 Videoma Archivo.....	11
1.5 Conclusiones.....	12
CAPÍTULO 2: “Herramientas y tecnologías actuales a desarrollar”	13
2.1 Introducción.....	13
2.2 Proceso Unificado de Desarrollo de Software (RUP) como base en el desarrollo de la solución propuesta.....	13
2.3 Lenguaje Unificado de Modelado (Unified Model Language UML)	15
2.4 Visual Paradigm for UML Enterprise Edition como herramienta para el modelado de la solución propuesta.....	16
2.5 Lenguajes de Programación.....	18
2.5.1 Lenguaje HTML	18
2.5.2 Lenguaje PHP	19
2.5.3 Lenguaje JavaScript	20
2.5.4 Lenguaje CSS	21
2.6 PostgreSQL como Sistema Gestor de Base de Datos a utilizar	21
2.7 Frameworks	23
2.7.1 Symfony.....	23
2.7.2 Dojo.....	24
2.8 NetBeans IDE 6.9 como Entorno de Desarrollo Integrado (IDE) a utilizar	25

2.9 Conclusiones.....	25
CAPÍTULO 3: “Presentación de la solución propuesta”	26
3.1 Introducción.....	26
3.2 Modelo de Negocio	26
3.2.1 Actores y Trabajadores del Negocio	26
3.2.2 Diagrama de Casos de Uso del Negocio	27
3.2.3 Descripción Textual de los Casos de Uso del Negocio	28
3.3 Requisitos del Sistema.....	37
3.3.1 Requisitos Funcionales.....	37
3.3.2 Requisitos no Funcionales.....	38
3.4.1 Descripción de los Actores del Sistema	40
3.4.2 Diagrama de Casos de Uso del Sistema.....	40
3.4.3 Descripción de los de los Casos de Uso del Sistema.....	41
3.5 Conclusiones.....	54
Capítulo 4: “Construcción de la solución propuesta”	55
4.1 Introducción.....	55
4.2 Arquitectura del Software	55
4.3 Patrones.....	56
4.3.1 Patrón de Arquitectura	56
4.3.2 Patrones de Diseño	57
4.4 Diagrama de Clases del Diseño	59
4.4.1 Diagrama de Clase del Diseño Gestionar Tarea.....	60
4.5 Principios de Diseño.....	61
4.6 Diseño de Base de Datos.....	61
4.7 Modelo de Despliegue.....	63
4.8 Modelo de Implementación	63
4.9 Pruebas del Sistema Propuesto	64
4.10 Conclusiones.....	68
CONCLUSIONES GENERALES	69
RECOMENDACIONES	70
GLOSARIO DE TÉRMINOS.....	71
BIBLIOGRAFÍA CITADA	72

INTRODUCCIÓN

Las Tecnologías de la Información y las Comunicaciones (TIC) son de gran relevancia y están presentes formando parte de la cultura tecnológica que rodea a la sociedad y con la que se ha acostumbrado a convivir. Estas han permitido ampliar nuestras capacidades físicas y mentales además de las posibilidades de desarrollo social.

El vertiginoso desarrollo de las ciencias informáticas a nivel mundial, ha tenido un gran impacto en el crecimiento de las TIC. Estas, a su vez, han permitido llevar la globalidad al mundo de la comunicación, proporcionando una mejor interconexión entre las personas y entidades a nivel mundial, y excluyendo barreras espaciales y temporales. No es menos cierto que en la actualidad, las TIC están sufriendo un desarrollo vertiginoso, y que esto está afectando prácticamente a todos los campos de nuestra sociedad, y la televisión no es una excepción. Este medio de comunicación audiovisual, que hasta hace poco más de 50 años era un privilegio para pequeños sectores de la sociedad, hoy cuenta con una alta demanda a nivel mundial, con emisores que presentan propuestas muy competitivas ofreciendo a la sociedad lo que esta quiere ver de forma muy sencilla. La televisión, como medio de comunicación social, ha permitido que la sociedad conozca todo lo referente al mundo que la rodea, ya sea de su mismo país o de cualquier otro, eliminando fronteras y aumentando su importancia para la educación y el proceso de aprendizaje de la actualidad.

Debido al avance de la tecnología, en varios países, muchos de los materiales audiovisuales que se transmiten por televisión o que en un momento fueron transmitidos, aún están almacenados en formato analógico. Esto implica que con el tiempo los materiales audiovisuales pierdan en calidad de imagen, sonido o ambos a la vez, ya sea por deterioro de los dispositivos de almacenamiento analógico u otras causas. Esta forma de almacenamiento de los materiales audiovisuales implica un freno para el desarrollo tecnológico porque en la actualidad, se está imponiendo en el mundo el formato digital, el mismo alarga la vida, mantiene la calidad y hace más portable la información, al contrario de los que se encuentran guardados en dispositivos de almacenamiento analógico, que muchos están obsoletos y otros casi discontinuados como es el caso de las cintas de celuloideas. La forma de almacenar los materiales audiovisuales en formato analógico al pasar de los años se vuelve más complejo, porque es necesario contar con un local que cumpla con estrictas condiciones que garanticen el perfecto estado de los mismos. Es complicado mantener estos materiales audiovisuales almacenados en

INTRODUCCIÓN.

dispositivos de formato analógico como las cintas de celuloideas y las variantes de las cintas magnéticas en perfectas condiciones porque son muy vulnerables a la humedad, al polvo, a la degradación del soporte además de que tienen un tiempo de conservación muy corto y su contenido se altera con facilidad.

El Instituto Cubano de Radio y Televisión (ICRT) no está exento de este problema ya que cuenta con una gran cantidad de materiales audiovisuales en formato analógico, muchos de los cuales poseen considerable importancia debido a su alto valor político, social y cultural, y que por tanto necesitan ser recuperados y digitalizados, en aras de conservarlos en el mejor estado posible.

Hace varios años en Cuba surgió, al albor de la batalla de ideas, la Universidad de las Ciencias Informáticas (UCI). La misma, aprovechando el desarrollo de las TIC, posee la capacidad de brindar servicios y desarrollar soluciones integrales de software. Exactamente en el proyecto de Sistema de Captura y Catalogación de Medias (SCCM) perteneciente al departamento de Señales Digitales del centro de desarrollo GEYSED de esta universidad, se desarrolla una aplicación informática con el fin de resolver el control, organización, catalogación, gestión y recuperación de los archivos de videos digitalizados en el ICRT y a su vez mejorar muchos de los procesos existentes en esta televisora. Todos los materiales audiovisuales del ICRT que se encuentran en formato analógico necesitan ser digitalizados para su mejor almacenamiento, conservación y acceso. Durante la digitalización se deben registrar las fallas que presenten dichos materiales para así coordinar las tareas asociadas a la ingesta del mismo en la aplicación informática desarrollada. Por estas razones, se determinó como **problema a resolver** para la investigación: ¿Cómo mejorar el control y la organización del proceso de gestión de fallas originadas durante la digitalización de materiales audiovisuales en el ICRT?

En la misma se determinó como **objeto de estudio** el proceso de gestión de fallas originadas durante la digitalización de materiales audiovisuales, estableciendo como **campo de acción** la informatización del proceso de gestión de fallas originadas durante la digitalización de materiales audiovisuales en el ICRT.

Para darle solución al problema a resolver anteriormente planteado, se propone como **objetivo general**: Desarrollar una aplicación informática que soporte el proceso de gestión de fallas originadas durante la digitalización de materiales audiovisuales en el ICRT.

INTRODUCCIÓN.

Como **idea a defender** se ha establecido: El desarrollo de una aplicación informática que soporte el proceso de gestión de fallas originadas durante la digitalización de materiales audiovisuales mejorará el control y la organización de este proceso en el ICRT.

Para lograr el cumplimiento del objetivo del presente trabajo de diploma se han propuesto las siguientes **tareas de investigación**:

- Caracterizar el proceso de gestión de fallas originadas durante la digitalización de materiales audiovisuales.
- Caracterizar a nivel nacional e internacional otras alternativas ya existentes que puedan dar solución al problema de la presente investigación.
- Seleccionar elementos técnicos que pueden ser incorporados a la solución propuesta a partir de las caracterizaciones anteriores.
- Justificar el uso de los métodos, herramientas y procedimientos más factibles para el desarrollo del componente.
- Definir los requisitos funcionales y no funcionales del componente.
- Diseñar el modelo de casos de uso del sistema.
- Desarrollar el modelo de diseño del sistema
- Implementar el componente.
- Diseñar pruebas para validar el cumplimiento de los requisitos funcionales y no funcionales.

Los principales resultados de esta investigación son: un subsistema web que soporte la gestión de fallas originadas durante la digitalización de materiales audiovisuales en el ICRT así como los artefactos y documentos generados durante el desarrollo y la actualización del estado del arte de los procesos de gestión de fallas originadas durante la digitalización de materiales audiovisuales.

Para adquirir conocimientos que permitan el cumplimiento de los objetivos propuestos en el desarrollo de la investigación se han seleccionado los siguientes métodos científicos de investigación, tanto teóricos como empíricos.

Los métodos teóricos utilizados durante la investigación fueron el análisis histórico-lógico, el analítico-sintético, inductivo-deductivo y la modelación.

INTRODUCCIÓN.

El método **análisis histórico-lógico** se utilizará con el fin de realizar un estudio analítico de la trayectoria histórica del objeto de estudio de la presente investigación, o sea, cómo se han desarrollado históricamente los procesos de gestión de fallas originadas durante la digitalización de materiales audiovisuales.

El **analítico-sintético** será usado con el objetivo de llegar a conocer, mediante el análisis de los diferentes documentos encontrados a lo largo de la investigación, la esencia de los fenómenos relacionados con el proceso de mejora del control y la organización del proceso de gestión de fallas originadas durante la digitalización de materiales audiovisuales en el ICRT.

El **inductivo-deductivo** se usará con la finalidad de generalizar el conocimiento adquirido acerca del control y la organización del proceso de gestión de fallas durante la digitalización de materiales audiovisuales, a partir de los aspectos tanto generales como particulares encontrados en la investigación y haciendo un análisis de lo particular a lo general y viceversa.

El método de **modelación** consiste en la creación de modelos para la reproducción del fenómeno que se estudia y hacer una representación simplificada de la realidad. Este método será usado para descubrir y estudiar nuevas relaciones y cualidades del objeto de estudio, y será aplicado fundamentalmente durante el proceso de desarrollo de software, en el cual se crean un grupo de modelos que describen los procesos acordes al desarrollo del componente.

Para el desarrollo de la investigación se han utilizado varios de los métodos empíricos, uno de ellos es el método de **observación**, que se usará con la finalidad de caracterizar y analizar en detalles cómo se realiza actualmente el proceso de gestión de fallas originadas durante la digitalización de materiales audiovisuales, identificando los mecanismos para desarrollar un componente acorde a las necesidades del Producto Sistema de Gestión de Audio, Video y Streaming para el ICRT.

Otro de los métodos utilizados es la **entrevista** que se utilizará con el fin de obtener información de gran importancia para el desarrollo del componente, mediante una conversación planificada entre el equipo de desarrollo y el cliente.

El presente documento está conformado de la siguiente manera:

Capítulo 1: Fundamentación teórica

En este capítulo se expone el estado del arte del objeto de estudio de la investigación así como los elementos teóricos que sustentan la misma, especificando algunos de los conceptos más importantes que permiten un mejor entendimiento del problema planteado. Se enuncian otras aplicaciones ya existentes que puedan dar cumplimiento de alguna manera al objetivo general planteado en la presente investigación y se explican las características fundamentales de los mismos.

Capítulo 2: Tendencias y tecnologías actuales a desarrollar

En este capítulo se justifica la selección de los lenguajes de programación que se van a utilizar así como la metodología de desarrollo de software, las herramientas y tecnologías para su posterior utilización en el desarrollo de la aplicación, analizando las características, ventajas y desventajas que justifiquen su uso.

Capítulo 3: Presentación de la solución propuesta

En este capítulo se describen los requisitos funcionales y no funcionales de la aplicación, además de los casos de uso identificados para la presente investigación. Se describe la solución propuesta a partir del modelo de negocio.

Capítulo 4: Construcción de la solución propuesta

En este capítulo se expone la descripción de los artefactos relacionados con el diseño e implementación del software que dará solución al problema a resolver del presente trabajo de diploma. Se realizarán las pruebas pertinentes para comprobar que la aplicación desarrollada cumple con los requisitos funcionales identificados con el cliente.

CAPÍTULO 1: “Fundamentación Teórica”.

1.1 Introducción

Para comprender en qué consiste la presente investigación, es necesario conocer todo lo relacionado con el objeto de estudio de la misma, en este caso: el proceso de gestión de fallas originadas durante la digitalización de materiales audiovisuales. En este capítulo se plantean conceptos y elementos teóricos que constituyen la base para el desarrollo de esta investigación, además de sustentar el entendimiento del problema científico y del objeto de estudio. Quedarán plasmados además, algunos de los aspectos teóricos fundamentales relacionados con el objeto de estudio y se analizarán otras aplicaciones existentes en el mundo que tienen entre sus funcionalidades la gestión de fallas originadas durante la digitalización de materiales audiovisuales.

1.2 Conceptos asociados al dominio del problema

Para un mejor entendimiento de lo planteado en la situación problemática y un mayor acercamiento a las características del proceso de gestión de fallas originadas durante la digitalización de materiales audiovisuales, es necesario conocer algunos conceptos asociados a la presente investigación.

La palabra **ingesta** posee varios significados, según estudios realizados se puede definir como la “Cantidad de sustancias o nutrientes ingeridos.” **(1)**, o “Acción y resultado de ingerir: Alimentos” **(2)**. En otra de las definiciones encontradas se denomina **ingesta** “al proceso de gestión digital en los archivos audiovisuales que captura y extrae los datos de sus contenidos en tiempo real. **Ingesta** en la producción televisiva de noticias se establece cuando existe un sistema de entrada de contenidos informativos por cualquier medio, cinta o línea, independientemente del programa al que vayan destinados y del uso que se les va a dar.” **(3)**

En el ámbito que define la presente investigación, se denomina **ingesta** al proceso de entrada, al Sistema de Captura y Catalogación de Medias, de los materiales audiovisuales en un formato final, o sea, ya digitalizados, que serán usados para su posterior organización, edición, catalogación, indexación, almacenamiento.

Otro de los conceptos a tratar es la palabra **proceso**, esta “tiene su origen del latín: processus. De acuerdo al diccionario de la Real Academia Española (RAE), el concepto hace referencia a la acción de ir hacia adelante, al transcurso del tiempo, al conjunto de las fases sucesivas de un fenómeno natural o de una operación artificial.” **(4)**

CAPÍTULO 1: “Fundamentación Teórica”.

Se puede determinar entonces que, un **proceso** no es más que un conjunto de acciones o eventos que se realizan o suceden para lograr un fin determinado.

La **gestión** “del latín *gestio: acción de administrar*. Es la actividad profesional destinada a establecer los objetivos y medios para la realización, organización del sistema, elaboración de la estrategia de desarrollo y ejecución de la gestión del personal. El término gestión, por lo tanto, implica al conjunto de trámites que se llevan a cabo para resolver un asunto o concretar un proyecto, se refiere al conjunto de actividades que desarrollan, movilizan y motivan al personal empleado que una empresa necesita para su éxito.” (4)

Se puede concluir entonces que **gestión** se refiere al conjunto de actividades que se realizan para establecer un objetivo, lograr un nivel de organización, establecer una estrategia, con el fin de resolver una cuestión determinada.

Se hace necesario esclarecer el significado de **materiales audiovisuales** ya que a estos términos se les atribuyen definiciones muy diversas. Como tal se puede plantear según estudios realizados que **audiovisual** se refiere “a las imágenes en movimiento y/o a los sonidos grabados, registrados en película, cinta magnética, disco o cualquier otro medio actualmente conocido o por inventar”. (5) “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... incluyen necesariamente también grabaciones sonoras”. (5)

Se define entonces que un **material audiovisual** es uno de los términos que se utilizan para referirse de forma más generalizada a los **audiovisuales**, o sea, a las imágenes en movimiento, sonidos grabados o ambos a la vez.

El proceso de **digitalizar** o **digitalización**, según estudios realizados, no es más que la “acción de convertir en digital (aportando valores en bits 0 y 1) cualquier tipo de información, ya sea gráfica, de audio, vídeo, etcétera”. (6) Se conoce también por la acción de “transformar una información a un sistema de dígitos para su tratamiento informático”. (7)

En este caso, **digitalizar** o **digitalización** se refiere a la acción que se lleva a cabo para convertir de un formato a otro, o sea, de analógico a digital.

1.3 Proceso de gestión de fallas originadas durante la digitalización de materiales audiovisuales.

Cuando un material audiovisual en formato analógico no se encuentra en buenas condiciones, o sea, su calidad no es buena, al ser digitalizado el audiovisual resultante no tendrá una máxima calidad. Para establecer una valoración de la calidad audio y video del material audiovisual, es de vital importancia la ocurrencia del proceso de gestión de fallas originadas durante la digitalización del mismo, que permitirá reportar y registrar todos los posibles errores que pueda tener este material.

1.3.1 Descripción general.

El proceso de gestión de fallas originadas durante la digitalización de materiales audiovisuales es un proceso de vital importancia para la caracterización de la calidad de los mismos. Después de la digitalización de un material audiovisual, es indispensable especificar su estado, o sea, en qué condiciones se encuentra esta calidad de video y de audio del material.

Por ejemplo, en un video que se encuentre en formato analógico la calidad de imagen es totalmente dependiente de la calidad de la cinta de video, el reproductor, u otros factores similares, en cambio, cuando el video es digital, la calidad de imagen es independiente y solo se ve afectada durante el proceso de digitalización. Los colores constituyen otra de las características a tener en cuenta debido a que en un video analógico se encuentran limitados los niveles de contraste y brillo al reproducir una imagen, característica que varía cuando el video es digitalizado, ya que cuando el video es digital, sus colores están basados en la combinación de los colores primarios. En fin, la calidad de un material audiovisual digitalizado dependerá de la calidad y del buen estado del video analógico del que proviene, por tal motivo es importante realizar un correcta gestión de las fallas de estos materiales digitalizados. Para lograr lo anteriormente planteado, el audiovisual sometido al proceso de digitalización será revisado críticamente y a medida que los errores sean identificados, tanto de audio como de video, serán registrados especificando el tiempo de comienzo y la duración de los mismos. Como resultado del proceso de gestión de fallas originadas durante la digitalización de los materiales audiovisuales, son creados reportes en los que se listan todas las fallas identificadas para cada material.

1.3.2 Descripción actual del dominio del problema.

El proceso de gestión de fallas originadas durante la digitalización de materiales audiovisuales es una de las tareas de gran escala que se realizan en el ICRT. Esto es debido a que los materiales

CAPÍTULO 1: “Fundamentación Teórica”.

audiovisuales que se encuentren en formato analógico, después de pasar por el proceso de digitalización (proceso que se lleva a cabo por digitalizadores externos), terminan teniendo características propiamente de ellos y es de suma importancia que se registren para su posterior uso. Como tal el proceso de gestión de fallas originadas durante la digitalización de materiales audiovisuales en el ICRT se realiza de manera manual, o sea, cuando un material audiovisual se va a digitalizar, existe una persona asignada que es la que se encarga de realizar esta operación. Esta persona es la responsable de realizar el registro de las fallas que puedan aparecer en el material digitalizado, registrándose las fallas encontradas en una hoja de papel. Esta descripción es asignada al video digitalizado para su posterior uso, ya sea su edición, descripción, organización, etc.

1.3.3 Situación problemática.

Debido al desenfrenado desarrollo de la tecnología, en varios países, muchos de los materiales audiovisuales que se transmiten por televisión aún están almacenados en formato analógico. Un caso específico de este problema es el ICRT, el cual cuenta con una inmensa cantidad de horas de materiales audiovisuales en este formato. Muchos de estos materiales audiovisuales se encuentran en peligro debido al tiempo que llevan almacenados en un lugar sin condiciones, donde se ven afectados por el polvo, la humedad, además de la magnetización de las cintas, etc.

Es de vital importancia para el ICRT la recuperación de estos materiales audiovisuales por el alto valor político, social y cultural que estos poseen para el país. Para ello el ICRT necesita digitalizar los materiales audiovisuales. Después de la digitalización se lleva a cabo el proceso de gestión de fallas originadas en la digitalización de estos materiales, proceso que se realiza manualmente por una persona asignada. Esta persona es la que reconoce las fallas que puedan aparecer en el video y las registra en una hoja de papel. Luego el reporte es asignado al material audiovisual que corresponde para su posterior uso, ya sea para ser almacenado o editado. Esta forma de gestionar los reportes de las fallas de los materiales audiovisuales no es eficiente debido a la forma de almacenamiento de los mismos. Esto puede traer consigo la pérdida del reporte de un material ya digitalizado, lo que provocaría que se tenga que realizar nuevamente el proceso de gestión de fallas. Además no se tiene una correcta descripción de los materiales audiovisuales para su posterior organización y búsqueda de los mismos. Puede darse el caso de que con la pérdida o extravío de un reporte, no se realice la correcta edición de un material audiovisual. Todos estos problemas traen consigo una mala organización de los materiales audiovisuales así como un mal control de los mismos, provocando que las búsquedas sean lentas y deficientes.

1.4 Análisis de otras soluciones existentes.

En el ámbito internacional, luego de investigar y consultar distintas fuentes, se encontró la existencia de otras aplicaciones relacionadas con el objetivo de la presente investigación y que son desarrolladas por empresas dedicadas a la producción de software. Las aplicaciones encontradas presentan funcionalidades similares a la solución de la situación problemática planteada en la presente investigación. Éstos software son privativos y de muy elevado costo, lo que dificulta su adquisición y por ende su utilización como solución al problema planteado.

1.4.1 Td Ficus.

Td Ficus es una herramienta privativa desarrollada por TEDIAL.

“TEDIAL (Tecnologías Digitales Audiovisuales), es una compañía de alta tecnología, ubicada en Málaga, España, especializada en la gestión de activos audiovisuales y procesos de control en sectores que incluyen broadcast, instituciones, sanidad e industria. Es líder del mercado en España y América Latina, globalmente activa, innovadora, completa y altamente funcional para los sistemas de gestión de procesos y de media”. **(8)**



Imagen 1: Interfaz de Td Ficus.

“Como herramienta principal de todo el trabajo se encuentra Ficus, el sistema de gestión de procesos de negocio (BPM) de Tedral. En este proyecto, Ficus será el que coordine todas las tareas asociadas a las ingestas de contenidos procedentes de digitalizadores externos y de la digitalización de contenidos dentro de las instalaciones de RTVE. Dentro de los flujos de trabajo orquestados por Ficus se incluye el registro de la entrada y salida de cintas, el control de la entrada de cintas procedentes de los procesos de transformación, el control de la recepción de los archivos MXF digitales, el segmentado,

CAPÍTULO 1: “Fundamentación Teórica”.

trimado y compactado de los clips individuales, la integración con los procesos de QC automáticos y manuales y su registro en el archivo Tarsys y en el sistema ARCA, el gestor documental de RTVE”. (9)
“Ficus automatiza estas tareas, ejecutando de forma automática y/o interactiva las tareas y organizando aquellas que requieren intervención humana de acuerdo a diferentes perfiles de operador. El uso del sistema BPM Ficus permite alcanzar el grado de eficiencia y seguridad en los flujos de trabajo que requiere un proyecto con el calibre del emprendido por RTVE para la preservación de sus fondos de vídeo”. (9)

Las características fundamentales que presenta esta herramienta son:

- ♣ “Asegura los procesos de producción de contenidos.
- ♣ Automatiza los movimientos de media entre subsistemas en la instalación.
- ♣ Asegura la disponibilidad de la media en cualquier puesto de trabajo en el formato apropiado.
- ♣ Integra información sobre la llegada de nuevo material desde el sistema de tráfico.
- ♣ Comprueba la localización de la media en dispositivos externos como video servidores”. (10)

1.4.2 Videoma Archivo

“ISID es una empresa española proveedora de software de gestión multimedia (Media Asset Management) y sistemas de monitoreo de medios. Desde 1996 lleva desarrollando aplicaciones para la gestión de vídeo, audio e imagen adaptada a diferentes mercados. Su producto principal es un MAM (Media Asset manager), robusto, modular y escalable. Gestiona los procesos de ingesta, catalogación, búsqueda y publicación de contenidos”. (11)

Dentro de la línea de productos de ISID se encuentra el software **Videoma Archivo**, el mismo “es la solución software para la gestión digital de vídeo, audio e imagen. Realiza la ingesta de contenido en cualquier formato, catalogación, gestión y publicación del mismo.

Diseñado para empresas o entidades que necesitan gestionar su videoteca, fonoteca o fototeca de una manera organizada e incluso distribuirla por Internet o a través de móviles.

El sistema es modular y escalable, ofreciendo una amplia gama de módulos que cumplen con las exigencias de diferentes usuarios. Videoma Archivo está diseñada para trabajar en red, a través de múltiples puestos, desde los que se pueden realizar consultas del material almacenado en el servidor central”. (12)

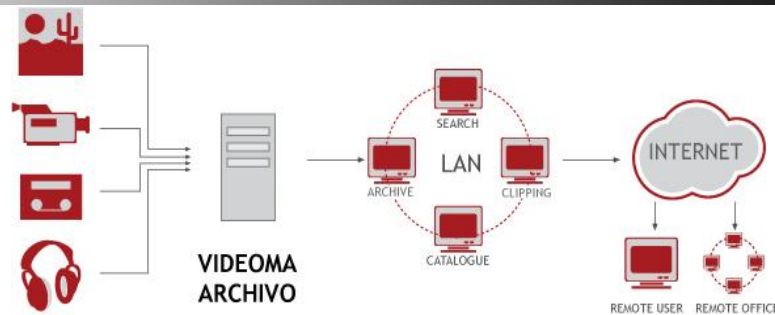


Imagen 2: Videoma Archivo.

“Para la importación de grandes cantidades de media **Videoma Archivo** cuenta con la herramienta Watchfolder que realiza la ingesta desatendida de múltiples archivos de vídeo, imagen o audio en el sistema. También permite la importación manual de archivos.” (12)

Videoma Archivo es una solución de software privativa que está funcionando desde el año 2006.

Las características fundamentales de este software son:

- ♣ “Su arquitectura WEB mejora los flujos de trabajo.
- ♣ Gestión completa de video, audio e imagen.
- ♣ Envío de vídeos a un editor profesional.
- ♣ Sistema abierto conformado por módulos con funcionalidades complementarias.
- ♣ Posibilidad de adicción de nuevos módulos desarrollados bajo la plataforma Videoma.
- ♣ Aplicación configurable en Español e Inglés.” (12)

1.5 Conclusiones.

En este capítulo se definieron aspectos fundamentales que permitieron entender el entorno en el que se encuentra enmarcada la presente investigación. Se plantean algunos conceptos necesarios para el mejor entendimiento del problema tales como: ingesta, proceso, gestión, entre otros. La realización de la descripción detallada del proceso de gestión de fallas permitió un mejor conocimiento y entendimiento de cómo se registran las mismas durante la digitalización de los materiales audiovisuales.

Con la descripción de las aplicaciones relacionadas con el objeto de estudio de la investigación, se pudo llegar a la conclusión de que no pueden ser utilizadas para darle solución al problema planteado en el presente trabajo de diploma debido a que son aplicaciones privativas y de muy elevado costo.

CAPÍTULO 2: “Herramientas y tecnologías actuales a desarrollar”.

2.1 Introducción

Actualmente existe en el mundo una gran variedad tecnológica que hace posible una gran gama de excelentes posibilidades para que empresas puedan dar soluciones efectivas a sus necesidades. Para darle solución al objetivo general de la presente investigación, se hizo preciso caracterizar las metodologías y herramientas existentes en la actualidad y determinar así las más factibles a utilizar en el proceso de desarrollo del producto deseado. Por tales motivos este capítulo tiene como objetivo caracterizar las herramientas y tecnologías seleccionadas con anterioridad por el arquitecto del proyecto en aras de dar cumplimiento de manera óptima a los requisitos identificados por el cliente y cumplir con las políticas de independencia tecnológica pautadas por nuestro país. Estará centrada entonces la exposición de ideas en cuanto las principales ventajas que aporta el uso de la tecnología elegida.

2.2 Proceso Unificado de Desarrollo de Software (RUP) como base en el desarrollo de la solución propuesta

Las metodologías de desarrollo de software surgen por la necesidad de utilizar una serie de procedimientos, técnicas, herramientas y soporte documental para el desarrollo de un producto de software, en cuyo ciclo de vida, se deben ir completando una serie de tareas, que siguiendo un grupo de fases y etapas, permiten obtener los componentes que dan lugar a un producto final. Cada una de estas tareas puede ser abordada y resuelta de disímiles maneras, con distintas técnicas y diversas herramientas. Es necesario saber cuándo culmina una tarea, quién debe realizarla, cuáles son las tareas que le anteceden o le preceden y la documentación utilizada para la realización de la misma. Se puede plantear entonces que en el desarrollo de un SW la metodología define quién está haciendo qué, cómo y cuándo.

El Proceso Unificado de Desarrollo de Software (RUP), nombrado así por sus siglas en inglés (Rational Unified Process), es una metodología que permite el desarrollo de aplicaciones a gran escala. Constituye unas de las metodologías estándar más utilizadas para el análisis, implementación y documentación de sistemas orientados a objetos. RUP es una de las más tradicionales, que se centra en la definición detallada de los procesos y tareas a realizar, propone una extensa documentación ya que pretende prever todo de antemano.

CAPÍTULO 2: “Herramientas y tecnologías actuales a desarrollar”.

RUP es una metodología robusta o pesada, que utiliza el Lenguaje Unificado de Modelado (UML) como lenguaje de representación visual y define “quién, cómo, cuándo y qué” debe hacerse en el proyecto. Cuenta con una forma organizada y disciplinada de asignar tareas y responsabilidades. Tiene como meta asegurar que la producción de software sea de alta calidad, que cumpla con las necesidades de los usuarios, dentro de las restricciones.

Se caracteriza principalmente por ser:

1. **Iterativo e incremental:** “El Proceso Unificado propone que cada fase se desarrolle en iteraciones, ya que el incremento en la complejidad de los sistemas actuales hace que sea factible dividir el trabajo en partes más pequeñas o mini-proyectos. Cada mini-proyecto es una iteración que resulta en un incremento. Una iteración involucra actividades de todos los flujos de trabajo, aunque desarrolla fundamentalmente algunos más que otros”. **(13)**
2. **Guiado por casos de uso:** “Los casos de uso representan los requisitos de software capturados durante el flujo de trabajo de requisitos, la planificación del proyecto se hace en términos de casos de uso, los desarrolladores crean realizaciones de casos de uso en términos de clases y subsistemas, los componentes se incorporan en los incrementos y cada uno realiza un conjunto de casos de uso, y por último se verifica que el sistema implementa los casos de uso correctos para el usuario. En otras palabras los casos de uso guían la arquitectura del sistema, enlazan todas las actividades del desarrollo y dirigen el proceso de desarrollo”. **(13)**
3. **Centrado en la arquitectura:** “La arquitectura representa la forma del futuro sistema en términos de vistas arquitectónicas, sobre la cual equipo de desarrollo y usuarios deben estar de acuerdo, ya que estas describen los elementos del modelo más importantes para su desarrollo, la arquitectura va madurando en las interacciones comenzando con los casos de uso relevantes desde el punto de vista arquitectónico”. **(13)**

Divide el desarrollo de SW en 4 fases. La fase de inicio que tiene como objetivo determinar la visión del proyecto. La fase de elaboración para determinar la arquitectura óptima. La fase de construcción, que tiene como objetivo lograr la capacidad operacional inicial del producto y la fase de transición, con el fin de desplegar el resultado del trabajo.

Es ideal para proyectos cuyos requisitos no son variables y para grandes equipos de desarrollo. Sin embargo puede adaptarse a diferentes condiciones. En RUP se han agrupado las actividades en

CAPÍTULO 2: “Herramientas y tecnologías actuales a desarrollar”.

grupos lógicos definiéndose 9 flujos de trabajo, los 6 primeros son flujos de ingeniería y los tres últimos de apoyo.

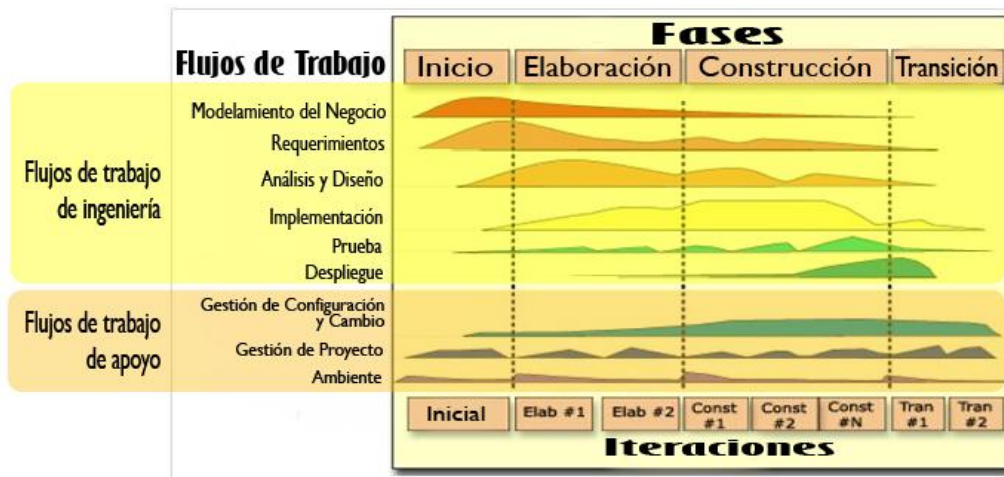


Imagen 3: Fases de RUP.

Se puede concluir entonces que RUP por ser la metodología que más se ajusta a las exigencias del producto a desarrollar. Es robusta y permite el desarrollo de software a gran escala y de larga duración. Genera abundante documentación apropiada para la comunicación entre los diferentes desarrolladores en equipos de trabajo grandes y que confiere ventajas en cuanto a la reutilización. Es una metodología configurable para satisfacer diferentes tipos de proyectos, está orientada al control de los procesos y establece rigor en las actividades a desarrollarse y las herramientas a utilizar.

2.3 Lenguaje Unificado de Modelado (Unified Model Language UML)

El Lenguaje Unificado de Modelado, también conocido como UML, fue construido hace más de una década por Grady Booch, Jim Rumbaugh e Ivar Jacobson. Estos tres reconocidos expertos en 1994 realizaron la primera versión de UML para la empresa Rational Software Corporation. En 1997, UML fue aceptado por la OMG (Object Management Group), una organización sin fines de lucro compuesta por destacadas compañías de la industria, como IBM, Sun Microsystems y HP, entre otras, que se encargan de la definición y mantenimiento de estándares para aplicaciones. **(14)**

UML es ante todo un lenguaje. Un lenguaje que proporciona un vocabulario y una reglas para permitir una comunicación. En este caso, este lenguaje se centra en la representación gráfica de un sistema. Este lenguaje indica cómo crear y leer los modelos.

CAPÍTULO 2: “Herramientas y tecnologías actuales a desarrollar”.

El lenguaje UML es un estándar OMG diseñado para visualizar, especificar, construir y documentar software orientados a objetos. El modelado en la construcción de software es de mucha importancia para lograr un mejor entendimiento de lo que se está haciendo. Permite descubrir oportunidades de simplificación y reutilización, ya que facilita los planos de un sistema y pueden ser detallados debido a que la modelación se basa en una representación visual y la combinación de diversos elementos gráficos que permiten la creación de diagramas. UML permite comunicar la estructura de un sistema complejo, y especificar el comportamiento deseado del mismo.

UML estandariza 9 tipos de diagramas para representar gráficamente un sistema desde distintos puntos de vista. Es muy utilizado en varias metodologías de desarrollo y presenta una estrecha relación con la escogida en el presente trabajo de diploma, debido a que RUP utiliza UML como lenguaje representación visual además de que utiliza todos los diagramas propuestos por este lenguaje. UML aunque es independiente del proceso de desarrollo, cuenta con su propia metodología de desarrollo denominada “Proceso Unificado de Desarrollo”, la cual está basada en componentes.

Los objetivos de UML son muchos, pero se pueden sintetizar sus funciones:

- **Visualizar:** UML permite expresar de una forma gráfica un sistema de forma que otro lo puede entender.
- **Especificar:** UML permite especificar cuáles son las características de un sistema antes de su construcción.
- **Construir:** A partir de los modelos especificados se pueden construir los sistemas diseñados.
- **Documentar:** Los propios elementos gráficos sirven como documentación del sistema desarrollado que pueden servir para su futura revisión. (15)

El lenguaje UML es perfecto para el modelado de sistemas orientados a objetos ya que incluye la representación de abstracciones, herencias, polimorfismos, encapsulamientos, envío de mensajes asociaciones y agregaciones. Permite detectar con facilidad las dependencias y dificultades implícitas del sistema. Se pueden modelar tanto sistemas de software como de hardware.

2.4 Visual Paradigm for UML Enterprise Edition como herramienta para el modelado de la solución propuesta

Las herramientas CASE son un conjunto de programas y ayudas que brindan asistencia técnica a analistas, ingenieros de software y desarrolladores para el análisis de requisitos, modelado visual y

CAPÍTULO 2: “Herramientas y tecnologías actuales a desarrollar”.

documentación durante parte o todo el ciclo de vida de un proyecto de software. CASE corresponde a las iniciales de Computer Aided Software Engineering que traducido al español significa: Ingeniería de Software Asistida por Computación. La selección de esta herramienta está estrechamente relacionada con la metodología de software y el lenguaje de modelado a utilizar.

Visual Paradigm para UML es una poderosa herramienta de fácil uso que soporta todo el ciclo de vida del desarrollo de software: análisis, diseño orientados a objetos, construcción, prueba y despliegue. Permite dibujar todos los tipos de diagramas de clases, permite realizar la conocida ingeniería inversa, es capaz de generar código a partir de diagramas realizados además de que permite generar documentación. “Es una herramienta que ofrece a los analistas del sistema todas las herramientas necesarias para capturar y organizar los requisitos.” **(16)** Brinda facilidad de integración con otras herramientas CASE y facilidades para la exportación e importación de componentes.

Visual Paradigm es capaz de integrarse con diversos IDE's como son: NetBeans (de Sun), JDeveloper (de Oracle), Eclipse (de IBM) y JBuilder (de Borland). Posee ingeniería inversa para: JAVA, .NET, XML e Hibernate.

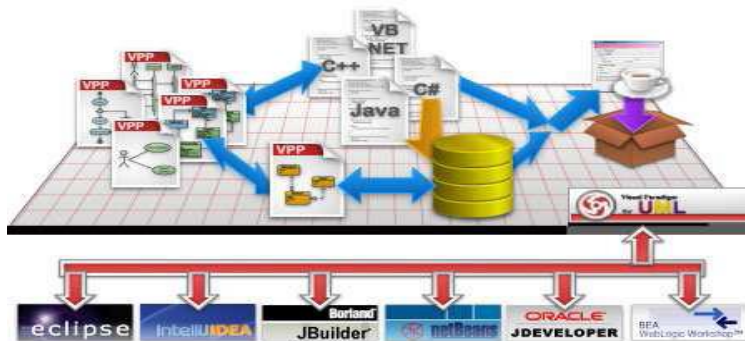


Imagen 4: Compatibilidad con Entornos de desarrollo y lenguajes.

Visual Paradigm se ha convertido en la herramienta de modelado más usada por excelencia para el desarrollo de sistemas en plataformas libres, que aunque posee una licencia comercial, se puede obtener una versión gratuita (licencia para Community Edition).

“Visual Paradigm como herramienta CASE ofrece:

- Entorno de creación de diagramas para UML 2.1.
- Diseño centrado en casos de uso y enfocado al negocio que generan un software de mayor calidad.
- Uso de un lenguaje estándar común a todo el equipo de desarrollo que facilita la comunicación.

CAPÍTULO 2: “Herramientas y tecnologías actuales a desarrollar”.

- Capacidades de ingeniería directa (versión profesional) e inversa.
- Modelo y código que permanece sincronizado en todo el ciclo de desarrollo.
- Disponibilidad de múltiples versiones, para cada necesidad.
- Disponibilidad de integrarse en los principales IDE's.
- Disponibilidad en múltiples plataformas”. **(17)**

Una característica fundamental que presenta esta herramienta CASE es la disponibilidad de múltiples plataformas, o sea, es soportada tanto en sistemas operativos Windows como en las distribuciones de GNU/Linux.

Es posible concluir entonces que la herramienta más apropiada a utilizar para el modelado de la presente investigación es Visual Paradigm para UML Enterprise Edition. Esta herramienta permite el modelado orientado a objetos y basado en el Lenguaje Unificado de Modelado (UML); es multiplataforma, posee gran disponibilidad de integración con varios entornos de desarrollos integrados que están bajo licencias de software libre como NetBeans. Es sencilla de instalar, fácil de utilizar y actualizar.

2.5 Lenguajes de Programación

“Un lenguaje de programación es aquel elemento dentro de la informática que nos permite crear programas mediante un conjunto de instrucciones, operadores y reglas de sintaxis; que pone a disposición del programador para que este pueda comunicarse con los dispositivos hardware y software existentes”. **(18)**

Existe a nivel mundial una amplia variedad de lenguajes de programación, entre ellos los lenguajes de programación web. Como el objetivo de la presente investigación es desarrollar el Módulo de Control de Ingesta para plataformas web, se han seleccionado para el desarrollo del presente trabajo de diploma, los lenguajes de programación web que a continuación se mencionan.

2.5.1 Lenguaje HTML

Desde el surgimiento de internet se han publicado sitios web gracias al lenguaje HTML. “Es un lenguaje estático para el desarrollo de sitios web (acrónimo en inglés de Hyper Text Markup Language, en español Lenguaje de Marcas Hipertextuales). Desarrollado por el World Wide Web Consortium (W3C). Los archivos pueden tener las extensiones (htm, html)”. **(19)**

CAPÍTULO 2: “Herramientas y tecnologías actuales a desarrollar”.

“El lenguaje HTML se basa en la sintaxis SGML (Standard Generalized Markup Language). Esto quiere decir que los diferentes elementos (párrafos, encabezamientos, tablas, listas) de un documento para la WWW se señalan intercalando etiquetas que indican al navegador cómo debe mostrarlo”. **(20)**

Dentro de las ventajas de html están:

- “Sencillo que permite describir hipertexto.
- Texto presentado de forma estructurada y agradable.
- No necesita de grandes conocimientos cuando se cuenta con un editor de páginas web o WYSIWYG.
- Archivos pequeños.
- Despliegue rápido.
- Lenguaje de fácil aprendizaje.
- Lo admiten todos los exploradores.” **(19)**

El lenguaje html cuenta con diversos elementos como son:

- Los caracteres.
- Marcadores, marcas o etiquetas.
- Nombre de etiquetas.
- Atributos.
- Comentarios.

2.5.2 Lenguaje PHP

Es un lenguaje de programación utilizado para la creación de sitio web. “PHP es un acrónimo recursivo que significa “PHP Hypertext Pre-processor”, (inicialmente se llamó Personal Home Page). Surgió en 1995, desarrollado por PHP Group”. **(19)**

El lenguaje PHP es un lenguaje de script interpretado del lado del servidor que se utiliza para la generación de páginas web dinámicas que son ejecutadas en el servidor. PHP no necesita ser compilado para que se ejecute. Sus archivos cuentan con la extensión (*.php).Es un lenguaje de código abierto.

Las ventajas que presenta este lenguaje son:

CAPÍTULO 2: “Herramientas y tecnologías actuales a desarrollar”.

- “Muy fácil de aprender.
- Se caracteriza por ser un lenguaje muy rápido.
- Soporta en cierta medida la orientación a objetos. Clases y herencia.
- Es un lenguaje multiplataforma: Linux, Windows, entre otros.
- Capacidad de conexión con la mayoría de los manejadores de base de datos: MySQL, PostgreSQL, Oracle, MS SQL Server, entre otras.
- Capacidad de expandir su potencial utilizando módulos.
- Posee documentación en su página oficial la cual incluye descripción y ejemplos de cada una de sus funciones.
- Es libre, por lo que se presenta como una alternativa de fácil acceso para todos.
- Incluye gran cantidad de funciones.
- No requiere definición de tipos de variables ni manejo detallado del bajo nivel.

PHP es un poderoso lenguaje e intérprete, ya sea incluido como parte de un servidor web en forma de módulo o ejecutado como un binario CGI separado, es capaz de acceder a archivos, ejecutar comandos y abrir conexiones de red en el servidor”. **(19)**

2.5.3 Lenguaje JavaScript

“Este es un lenguaje interpretado, no requiere compilación. Fue creado por Brendan Eich en la empresa Netscape Communications. Utilizado principalmente en páginas web. Es similar a Java, aunque no es un lenguaje orientado a objetos, el mismo no dispone de herencias. La mayoría de los navegadores en sus últimas versiones interpretan código JavaScript”. **(19)**

El lenguaje JavaScript puede ser integrado dentro de las páginas web. La World Wide Web diseñó un estándar denominado Modelo de Objetos del Documento (DOM) para evitar incompatibilidades. JavaScript es un lenguaje interpretado orientado a pseudo-código debido a que no existe herencia.

Algunas de las ventajas que presenta son:

- “Código visible por cualquier usuario.
- El código debe descargarse completamente.
- Puede poner en riesgo la seguridad del sitio, con el actual problema llamado XSS (significa en inglés Cross Site Scripting renombrado a XSS por su similitud con las hojas de estilo CSS)” **(19)**.

CAPÍTULO 2: “Herramientas y tecnologías actuales a desarrollar”.

2.5.4 Lenguaje CSS

CSS son las siglas de Cascading Style Sheets, que en español significa Hojas de Estilo en Cascadas. Es un lenguaje que describe la presentación de los documentos estructurados en hojas de estilos para diversos métodos de interpretación, o sea, describe cómo se va a mostrar un documento en pantalla. El CSS es utilizado para dar estilo a documentos HTML, XML, XHTML, SVG o XUL, separando el contenido de la presentación del documento. Esta incluye elementos tales como los colores, fondo, márgenes, bordes, tipo de letra...modificando la apariencia de una página web de manera más sencilla. CSS permite a los desarrolladores web controlar el estilo y formato de múltiples páginas web al mismo tiempo y cualquier cambio en el estilo marcado para un elemento en la CSS afectará a todas las páginas vinculadas a éste.

El lenguaje de las hojas de estilo está definido en las especificaciones CSS1 y CSS2 del World Wide Web Consortium (W3C). Es un estándar aceptado por toda la industria relacionada con la web, o por lo menos por la gran mayoría de los navegadores, aunque estos no implementan las funcionalidades del CSS de la misma manera.

Se puede asociar las reglas de estilos CSS a las marcas HTML de tres maneras:

- Directamente a la marca.
- En el head de la página.
- Agrupando las reglas de estilo en un archivo independiente con extensión *.css.

2.6 PostgreSQL como Sistema Gestor de Base de Datos a utilizar

Un Sistema Gestor de Base de Datos (SGBD) no es más que el conjunto de programas para administrar y gestionar la información contenida en una base de datos. Los SGBD permiten la definición y creación de una base de datos, la manipulación de los mismos, un acceso controlado a los datos y el acceso compartido a la base de datos. Es de suma importancia que aseguren la integridad, confiabilidad y seguridad de los datos.

Entre los SGBD más conocidos y usados actualmente se encuentran: MySQL, Oracle y PostgreSQL. En este caso se expondrán las principales características y ventajas de este último, por ser el seleccionado por el arquitecto del proyecto para ser usado en el producto. PostgreSQL es considerado un gestor de base de datos relacional orientado a objetos y de código abierto, liberado bajo la licencia BSD (Berkeley Software Distribution), que en español significa Distribución de Software Berkeley, la que permite el uso de código fuente de software no libre. Incorpora cuatro conceptos adicionales que

CAPÍTULO 2: “Herramientas y tecnologías actuales a desarrollar”.

permiten que los usuarios puedan fácilmente extender el sistema: clase, herencia, tipos y funciones. Otras de las características que ofrecen potencia y flexibilidad a este gestor son las restricciones, disparadores, reglas y la integridad transaccional. Gracias a estas características es considerado como una base de datos objeto-relacional. El desarrollo de PostgreSQL no es controlado por una compañía, sino que descansa en una comunidad de desarrolladores y varias compañías. Cumple con los estándares SQL (los ANSI-SQL 92/99 y se adapta a los estándares SQL 2003), es multiplataforma y está diseñado para ambientes de alto volumen. Es considerado el SGBD de código abierto más potente del mercado. Se basa en el modelo cliente/servidor. El lenguaje procedimental propio es el PL/pgsql y extendido el PL/perl, PL/PYTHON, PL/JAVA, PL/RUBY, entre otros.

Los límites que presenta PostgreSQL son:

1. Máximo de la base de datos: ilimitado.
2. Máximo de tablas: 32TB.
3. Máximo de tupla: 1.6TB.
4. Máximo de campo: 1GB.
5. Tablas x tabla: ilimitado.
6. Índices por tabla: ilimitado.
7. Columnas máximas por tabla 250 – 1600 dependiendo del tipo.

PostgreSQL presenta varias herramientas gráficas de alta calidad para administrar bases de datos como el pgAdmin y el pgExplorer. Para la realización de los diseños de las base de datos cuenta con el Tora y el Data Architect. Puede conectarse por ODBC, OLE DB y JDBC, además de los conectores propios de cada lenguaje. Se maneja como un servicio del sistema, levanta memoria compartida, vigila solicitudes y está al tanto de los movimientos, puede manejar varias bases de datos y usuarios.

“Como base de datos empresarial, PostgreSQL se enorgullece de características sofisticadas como el Control de Concurrencia Multi-versión (MVCC), recuperación a punto, espacios de tablas, replicado asíncrono, transacciones anidadas (con puntos de guardado), backups online o en caliente, un planificador/optimizador sofisticado de consultas y un sistema de logs con escritura adelantada para mayor tolerancia a fallos”. **(21)**

Luego de analizar sus características se concluye que es la mejor opción pues además de ser orientado a objetos y de código abierto, es capaz de ajustarse a la cantidad de memoria que posea el sistema a diferencia de otros gestores de base de datos. Permite el uso de herencia y polimorfismo. Cumple completamente con ACID. Corre sobre todos los sistemas operativos mayoritarios incluyendo

CAPÍTULO 2: “Herramientas y tecnologías actuales a desarrollar”.

Linux, UNIX (AIX, BSD, HP-UX, SGI IRIX, Mac OS X, Solaris, Tru64) y Windows. Soporta el almacenamiento de archivos binarios grandes, incluyendo imágenes, sonido o video. Además presenta interfaces de programación nativas para C/C++, Java, Perl, Python, Ruby, Tcl, ODBC. Es altamente escalable un cuanto a la cantidad de datos y usuarios.

2.7 Frameworks

Un framework de desarrollo es una importante herramienta para el desarrollo y/o implementación de una aplicación en la actualidad. Es una estructura tecnológica compuesta por librerías, componentes y clases que facilitan el desarrollo ágil, seguro y escalable de una aplicación. Proporciona tanto a los desarrolladores como a los diseñadores una mejor organización y estructura de sus proyectos, acelerando el desarrollo del producto final. La utilización del framework adecuado puede proporcionar patrones que se utilizan para resolver tareas comunes, encapsulando operaciones complejas en instrucciones sencillas. Se puede determinar que los framework se encargan de los pequeños detalles permitiendo que los programadores se dediquen a las partes que realmente importan.

2.7.1 Symfony

Symfony es un framework PHP de código libre con programación orientada a objetos que hace más sencilla la creación de aplicaciones web con PHP. Proporciona una arquitectura, componentes y herramientas para desarrolladores que le permiten crear aplicaciones web complejas de manera más rápida. La utilización de este framework de desarrollo permite la creación de sitios web organizados permitiendo que la realización de la documentación del proyecto sea más sencilla.

Algunas de las características que presenta este framework son:

- “Fácil de instalar y configurar en sistemas Windows, Mac y Linux.
- Funciona con todas las bases de datos comunes (MySQL, PostgreSQL, SQLite, Oracle, MS SQL Server).
- Compatible solamente con PHP 5 desde hace años, para asegurar el mayor rendimiento y acceso a las características más avanzadas de PHP.
- Basado en la premisa de "convenir en vez de configurar", en la que el desarrollador solo debe configurar aquello que no es convencional.
- Preparado para aplicaciones empresariales, ya que se puede adaptar con facilidad a las políticas y arquitecturas propias de cada empresa u organización.
- Flexible hasta cualquier límite y extensible mediante un completo mecanismo de plugins.

CAPÍTULO 2: “Herramientas y tecnologías actuales a desarrollar”.

- Publicado bajo licencia MIT de software libre y apoyado por una empresa comprometida con su desarrollo.
- Traducido a más de 40 idiomas y fácilmente traducible a cualquier otro idioma”. **(22)**

Es un framework de desarrollo que cuenta con una abundante documentación ya que cuenta con cinco libros publicados que son gratuitos de calidad y siempre actualizados. Puede ser utilizado tanto en pequeños como en grandes proyectos. Proporciona un aumento de la productividad ayudando a mejorar la calidad de las aplicaciones web aplicando todas las buenas prácticas y patrones de diseños que se han definido para la web. Es infinitamente escalable para programar aplicaciones con una gran cantidad de usuarios. Symfony es un framework que está publicado bajo la licencia MIT, que permite desarrollar aplicaciones web comerciales, gratuitas o de software libre. Permite controlar el acceso a la información e incluye por defecto protección contra ataques XSS y CSRF.

2.7.2 Dojo

Dojo es un framework de desarrollo que contiene APIs y widgets (controles) para facilitar el desarrollo de aplicaciones web que utilicen tecnología AJAX. Contiene un sistema de empaquetado inteligente, los efectos de UI, drag and drop APIs, widget APIs, abstracción de eventos, almacenamiento de APIs en el cliente e integración de APIs con AJAX. Dojo resuelve asuntos de usabilidad comunes como pueden ser la navegación y detección de navegador, soportar cambios de URL en la barra de URL para luego regresar a ellas (bookmarking) y la habilidad de degradar cuando AJAX/JavaScript no es completamente soportado por el cliente. Está escrito en el lenguaje de programación JavaScript y permite a los desarrolladores de aplicaciones web usar funcionalidades y efectos dinámicos a las páginas que programen. “Según estudios realizados, se ha obtenido información de que dojo es el framework JS más rápido seguido de MooTools 1.2.1 y jQuery 1.3.2”. **(23)**

Algunas de las características que presenta este framework son:

- “Dojo resuelve asuntos de usabilidad comunes como pueden ser la navegación y detección del navegador.
- Dojo dispone de varios componentes entre ellos: Menús, pestañas, tooltips, tablas ordenables, gráficos dinámicos y dibujados de vectores 2D.
- Se pueden crear animaciones personalizables.
- Soporte para arrastrar y soltar.
- Formularios y rutinas de validación para los parámetros.

CAPÍTULO 2: “Herramientas y tecnologías actuales a desarrollar”.

- Editor online de texto enriquecido.
- Ahorra el tiempo de desarrollo y diseño de interfaces complicadas y manejos de eventos complejos”. (24)

2.8 NetBeans IDE 6.9 como Entorno de Desarrollo Integrado (IDE) a utilizar

NetBeans no es más que un entorno de desarrollo integrado (IDE) de código abierto que Sun Microsystems creó en junio del 2000. Es una herramienta que permite a los programadores escribir, depurar y ejecutar programas. Está escrito en Java, pero puede servir para cualquier otro lenguaje de programación. NetBeans es un producto libre y gratuito sin restricciones de uso, lo que lo hace muy conveniente para el desarrollo del producto final de la presente investigación. Permite la creación de aplicaciones tanto de escritorio como web. Es posible su uso en diversos sistemas operativos como: Windows, GNU/Linux, Solaris, Mac OS, lo que lo convierte en un IDE multiplataforma. NetBeans proporciona una estructura para los proyectos que podamos crear proponiendo un esqueleto para la organización del código. Permite la integración de los lenguajes de programación web como son HTML, JavaScript y CSS. Permite la integración con diversos framework de desarrollo entre los que están los framework de PHP como Symfony. Presenta un editor de PHP ágil y robusto, reconocimiento de sintaxis y lo que provee la última versión de PHP, la 5.3.

2.9 Conclusiones

Durante el desarrollo del presente capítulo se valoró lo importante que es para el desarrollo del presente trabajo de diploma usar herramientas bajo licencias de software libre, para el desarrollo de una solución informática, lo cual fue muy determinante a la hora de seleccionar una. Por ejemplo, se seleccionó como herramienta CASE el Visual Paradigm debido a la versión community de su licencia. En el caso de la selección de los SGBD, se escogió PostgreSQL debido a su licencia BSD y las características que posee.

Las herramientas se han escogido, no porque una sea mejor que la otra, sino porque según sus características, las escogidas son las que más se ajustan a las exigencias de la presente investigación.

CAPÍTULO 3: “Presentación de la solución propuesta”.

3.1 Introducción

Según el trabajo de ingeniería que propone RUP, lo que más peso tiene en la fase de inicio son los flujos de trabajo Modelamiento del Negocio y Requisitos. El objetivo de este capítulo es realizar la representación de los principales artefactos obtenidos durante el desarrollo de la solución propuesta. Del modelo de negocio: los actores, trabajadores, así como una descripción de los mismos, además el diagrama de casos de uso del negocio, sus descripciones textuales y los diagramas de actividades correspondientes para cada caso. En cuanto al flujo de trabajo de Requisitos se enunciarán los requisitos funcionales y no funcionales y se representará el diagrama de casos de uso del sistema, acompañado de sus respectivas descripciones.

3.2 Modelo de Negocio

Durante el proceso de desarrollo de software es muy necesario el estudio de la estructura y los procesos que serán automatizados por la aplicación que se va a construir. La modelación del negocio permite obtener una visión de la organización o empresa con el fin de conocer sus procesos así como determinar los requisitos del sistema. Permite describir los procesos del negocio, identificando quienes participan y las actividades que requieren automatización. Permite encontrar los actores y casos de uso del negocio así como la construcción del modelo de casos de uso del negocio, detallar estos casos de uso, encontrar entidades y trabajadores, construir el modelo de análisis, etc. En fin, el objetivo del modelo de negocio es describir los procesos identificados con el propósito de comprenderlos.

3.2.1 Actores y Trabajadores del Negocio

Un actor del negocio es cualquier individuo, grupo, entidad, organización, máquina o sistema de información externos con los que el negocio interactúa. Lo que se modela es el rol que cumple cuando interactúa con el negocio para obtener beneficios de sus resultados.

Un trabajador del negocio representa un rol que juega una persona o grupo de personas, máquina o un sistema automatizado que actúan en el negocio. Estos son los que realizan las actividades interactuando con otros trabajadores del negocio y manipulando entidades.

CAPÍTULO 3: “Presentación de la solución propuesta”.

Actor	Descripción
Coordinador de archivología	Persona encargada de supervisar y controlar todas las actividades que se realizan sobre el archivo audiovisual. Es el encargado de asignar las tareas de digitalización, edición y descripción.

Tabla 1: Actores del negocio y sus descripciones.

Trabajador	Descripción
Digitalizador	Persona encargada de digitalizar los materiales audiovisuales y registrar las fallas ocurridas durante la digitación.
Editor	Persona encargada de editar los materiales audiovisuales y eliminar las fallas reportadas durante la digitalización. Además de describirlos brevemente, teniendo en cuenta el contenido y equipo de realización de los mismos.
Supervisor	Persona encargada de evaluar la calidad de los materiales, generalmente interviene luego de la edición, para comprobar la efectividad de este proceso.
Diseñador	Persona que encargada de diseñar imágenes identificativas de cada material digitalizado (para posters, portadas de CDs, etc.)

Tabla 2: Trabajadores del negocio y sus descripciones.

3.2.2 Diagrama de Casos de Uso del Negocio

El diagrama de casos de uso del negocio es la representación gráfica de un conjunto de elementos, dígame actores y casos de uso, así como las relaciones y dependencias que se establecen entre ellos. En estos aparece cada proceso del negocio como un caso de uso. El diagrama de caso de uso del negocio permite a los desarrolladores describir cómo el negocio es utilizado por sus clientes y socios.

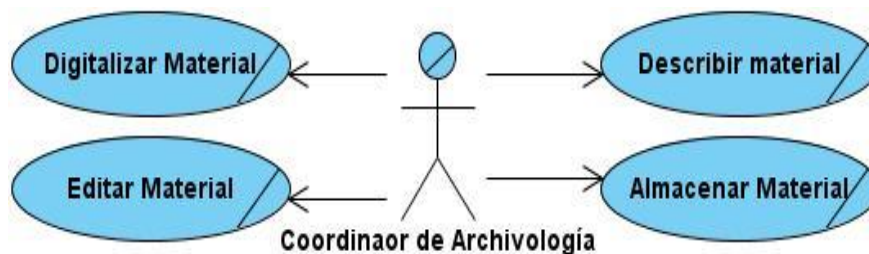


Imagen 5: Diagrama de Casos de Uso del Negocio.

CAPÍTULO 3: “Presentación de la solución propuesta”.

3.2.3 Descripción Textual de los Casos de Uso del Negocio

Descripción Textual del Casos de Uso del Negocio Digitalizar Material

Caso de Uso del Negocio	Digitalizar Material
Actores	Coordinador de Archivología
Resumen	El caso de uso se inicia cuando el Coordinador de Archivología ordena la digitalización de un material audiovisual en específico y le asigna esta tarea a un Digitalizador. Este último, mientras digitaliza el material registra en un documento todas las fallas (de audio y video) detectadas durante este proceso. El caso de uso termina cuando se obtiene una copia digital del material seleccionado, así como una lista de las fallas detectadas.
Casos de uso asociados	
Flujo Normal de los Eventos	
Acción del actor	Respuesta del proceso de negocio
1. El Coordinador de Archivología asigna a un Digitalizador la tarea de digitalizar un material audiovisual.	1.1 El Digitalizador digitaliza el material audiovisual con la ayuda de tecnología especializada, visualizando el material mientras dura este proceso.
	1.2 Registra manualmente, en el Registro de Fallas, las fallas (de audio y video) que se detecten durante la digitalización del material audiovisual.
	1.3 El Digitalizador realiza una evaluación del material determinando si el material es utilizable o no.
	1.4 Si el material audiovisual es declarado como utilizable, luego de concluida la digitalización, se guarda en un servidor para materiales a editar (servidor de edición).
	1.5 El Digitalizador le Informa al Coordinador de

CAPÍTULO 3: “Presentación de la solución propuesta”.

	Archivos que la copia digital del material audiovisual está lista para ser editada.
Flujo Alterno 1	
Acción del actor	Respuesta del proceso de negocio
	1.4 Si el material audiovisual es declarado como no utilizable, el Digitalizador informa al Coordinador de Archivología la pérdida de los materiales almacenados en el contenedor.
Mejoras propuestas	

Tabla 3: Descripción textual del CUN Digitalizar Material.

Diagrama de Actividades del Caso de Uso del Negocio Digitalizar Material

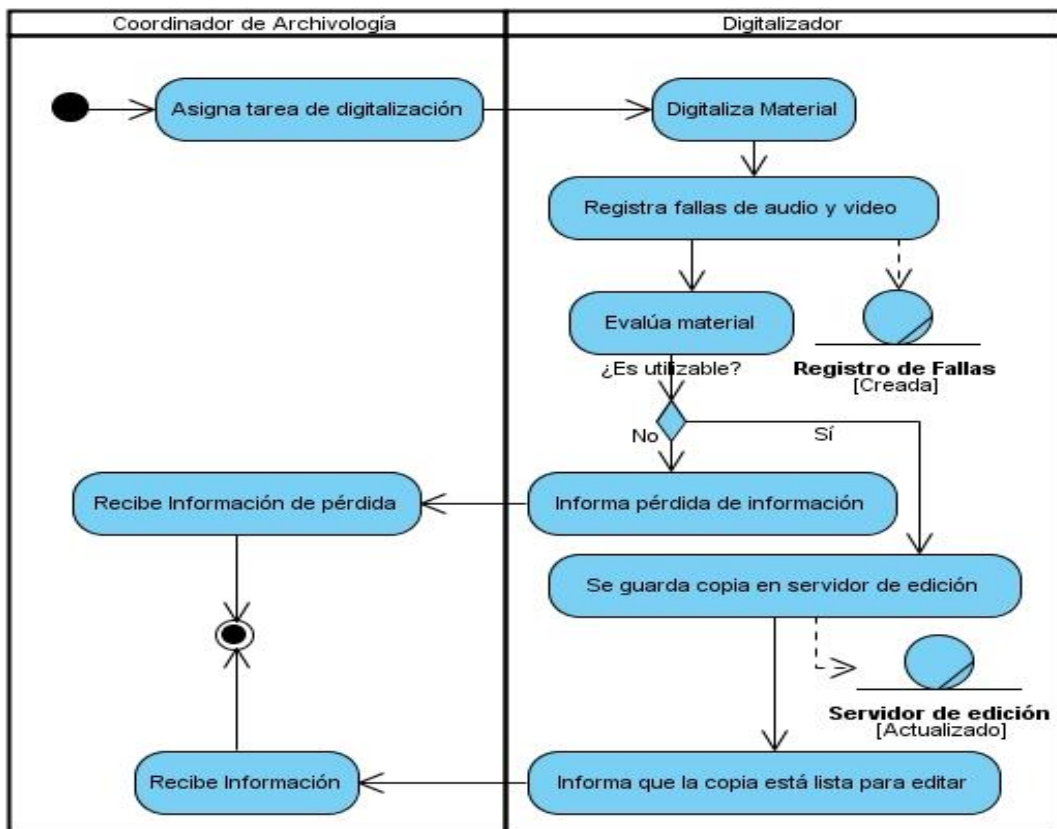


Imagen 6: Diagrama de Actividades del Caso de Uso del Negocio Digitalizar Material.

CAPÍTULO 3: “Presentación de la solución propuesta”.

Descripción Textual del Caso de Uso del Negocio Editar Material

Caso de Uso del Negocio	Editar Material
Actores	Coordinador de Archivología
Resumen	El caso de uso comienza cuando el Coordinador de Archivología asigna la edición de un material audiovisual (en formato digital) al cual le fueron detectadas fallas durante su digitalización. El editor consulta con detenimiento el listado de las fallas generadas como resultado del caso de uso del negocio “Digitalizar Material” (Registro de Fallas) y soluciona la mayor cantidad de fallas posibles y el caso de uso concluye con la corrección de las fallas y el material audiovisual editado.
Casos de uso asociados	
Flujo Normal de los Eventos	
Acción del actor	Respuesta del proceso de negocio
1. El Coordinador de Archivología asigna a un Editor la tarea de editar un material audiovisual, para ello le proporciona la dirección URL del mismo y el correspondiente Registro de Fallas.	1.1. El Editor descarga el video desde la dirección URL proporcionada.
	1.2. El Editor elimina la mayor cantidad de fallas posibles del material audiovisual basándose en el Registro de Fallas.
	1.3. Una vez concluida la edición del material audiovisual seleccionado, y eliminadas la mayor cantidad de fallas posibles, el Editor le Informa al Coordinador de Archivología que concluyó el proceso de edición.
2. El Coordinador de Archivología asigna a	2.1. El Supervisor Visualiza el material

CAPÍTULO 3: “Presentación de la solución propuesta”.

<p>un Supervisor la tarea de evaluar la calidad del material editado.</p>	<p>audiovisual editado y lo evalúa. Consulta para ello el registro de fallas en aras de detectar si fueron resueltas todas las fallas listadas anteriormente.</p>
	<p>2.2. Si el resultado de la evaluación del material audiovisual editado es satisfactorio, entonces este está listo para ser descrito y por tanto se guarda en un servidor para materiales a describir (servidor de descripción).</p>
	<p>2.3. El Supervisor le informa al Coordinador de Archivología que la copia digital del material audiovisual está lista para ser descrita y se encuentra copiada ya en el servidor de descripción.</p>
<p>Flujo Alterno 1</p>	
<p>Acción del actor</p>	<p>Respuesta del proceso de negocio</p>
	<p>2.2. Si el resultado de la evaluación del material audiovisual editado no es satisfactorio, el Supervisor actualiza el Registro de Fallas señalando aquellas que aún persisten.</p>
	<p>2.3. El Supervisor le informa al Coordinador de Archivología las fallas que aún persisten.</p>
<p>3. El Coordinador de Archivología asigna a un Editor la tarea de reeditar el material audiovisual evaluado, para ello le proporciona la dirección URL del mismo y el correspondiente Registro de Fallas (actualizado).</p>	<p>3.1. Se retoma el flujo normal de los eventos a partir de la acción 1.1.</p>
<p>Mejoras propuestas</p>	

Tabla 4: Descripción textual del CUN Editar Material.

CAPÍTULO 3: “Presentación de la solución propuesta”.

Diagrama de Actividades del Caso de Uso del Negocio Editar Material

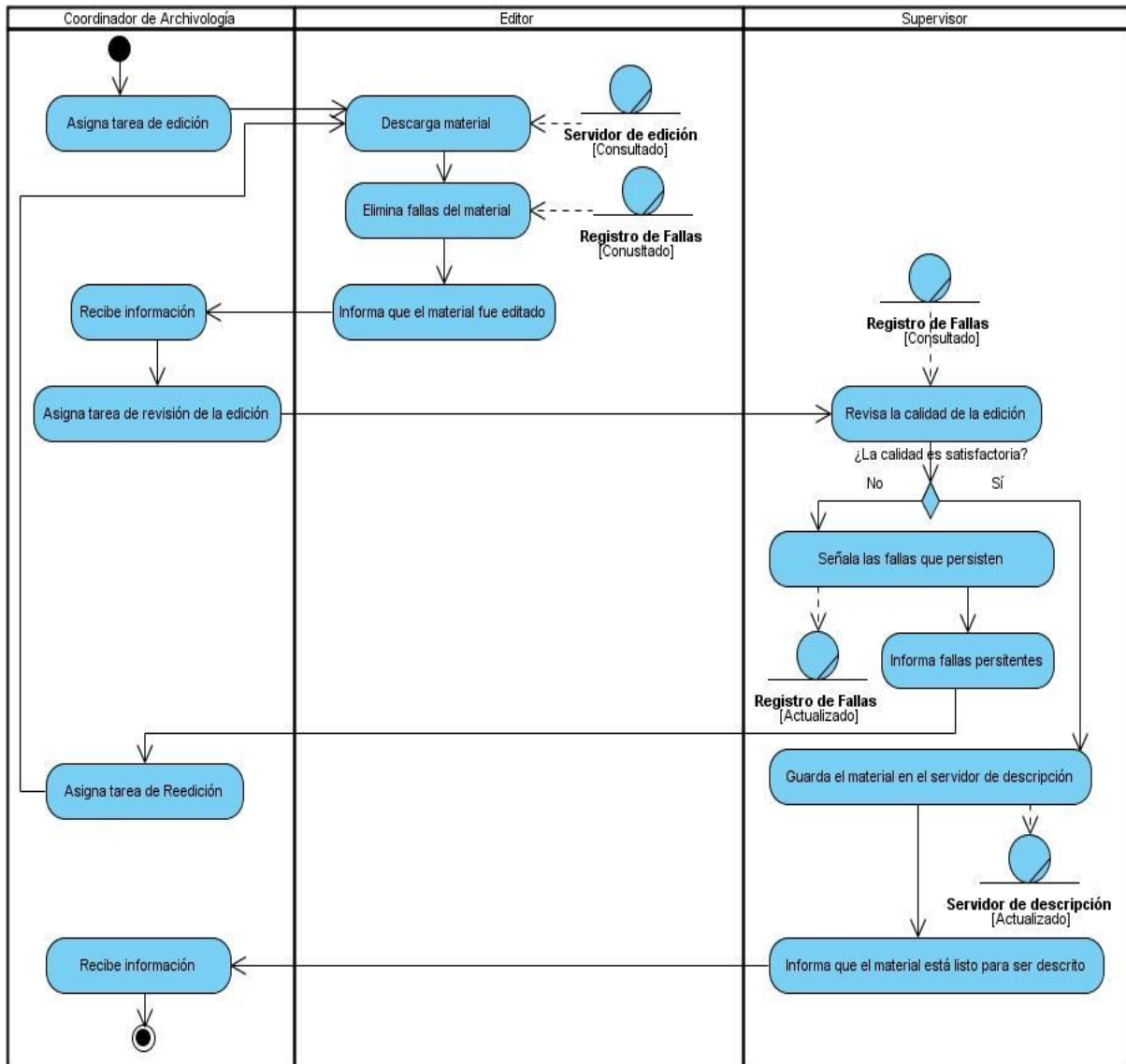


Imagen 7: Diagrama de Actividades del Caso de Uso del Negocio Editar Material.

Descripción Textual del Caso de Uso del Negocio Describir Material

Caso de Uso del Negocio	Describir Material
Actores	Coordinador de Archivología
Resumen	El caso de uso comienza cuando el Coordinador de Archivología asigna la descripción de un material audiovisual (en formato digital) el cual ya

CAPÍTULO 3: “Presentación de la solución propuesta”.

	fue editado correctamente y cuya calidad fue certificada por el supervisor. El material es descrito y es registrada esta información en el Catálogo.
Casos de uso asociados	
Flujo Normal de los Eventos	
Acción del actor	Respuesta del proceso de negocio
1. El Coordinador de Archivología asigna a un Editor la tarea de describir un material audiovisual (en formato digital), para ello le proporciona la dirección URL del mismo.	1.1. El Editor copia el video desde la dirección URL proporcionada.
	1.2. El Editor visualiza el material descargado y le asigna un número identificativo.
	1.3. El Editor registra en el “Catálogo” una descripción del material. Esta descripción incluye, entre otros datos: <ul style="list-style-type: none"> • Nombre del material. • Clasificación. • Equipo de realización. • Elenco de actores. • Fecha de realización. • Resumen breve. Nota: se hace necesario exponer que en ocasiones estos datos varían en dependencia de la clasificación del material, pues, por ejemplo, no se guarda la misma información para un programa deportivo que para una película.
	1.4. El Editor Informa al Coordinador de Archivología que el material está descrito.
Mejoras propuestas	

Tabla 5. Descripción textual del CUN Describir Material.

CAPÍTULO 3: “Presentación de la solución propuesta”.

Diagrama de Actividades del Caso de Uso del Negocio Describir Material

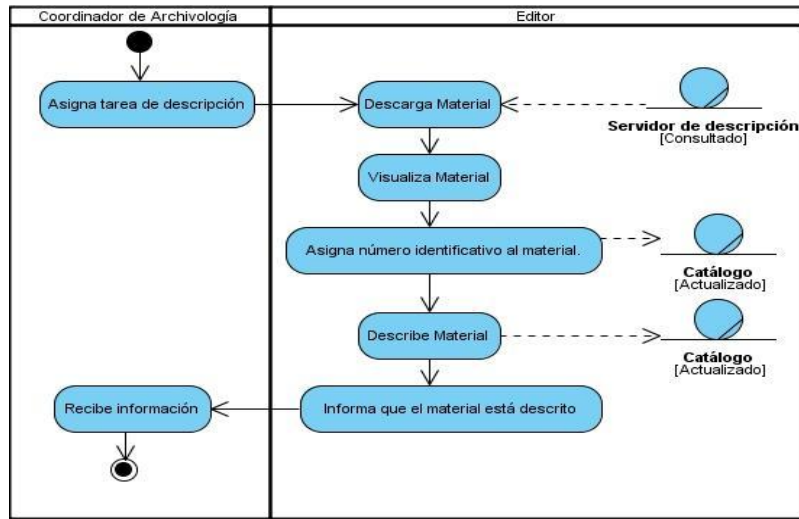


Imagen 8: Diagrama de Actividades del Caso de Uso del Negocio Describir Material.

Descripción Textual del Caso de Uso del Negocio Almacenar Material

Caso de Uso del Negocio	Almacenar Material
Actores	Coordinador de Archivología
Resumen	El caso de uso comienza cuando el Coordinador de Archivología asigna la tarea a un Supervisor de guardar un material audiovisual ya descrito. El Supervisor guarda en un disco duro una copia del material audiovisual descrito y otra copia en un contenedor (CD o DVD). Como resultado se actualiza el catálogo con el código del disco duro y el contenedor en el que se guardó cada material y estos se almacenan en el Archivo.
Casos de uso asociados	
Flujo Normal de los Eventos	
Acción del actor	Respuesta del proceso de negocio
1. El Coordinador de Archivología asigna a un Supervisor la tarea de almacenar un material audiovisual (en formato digital), para ello le proporciona la dirección URL	1.1. El Supervisor descarga el material desde la dirección URL proporcionada.

CAPÍTULO 3: “Presentación de la solución propuesta”.

del mismo y el número identificativo correspondiente al material en el Catálogo.	
	<p>1.2. El Supervisor asigna a un Diseñador la tarea de diseñar una portada para el material a almacenar. Para ello le hace llegar una “Petición de Diseño” documento con los datos necesarios para diseñar la portada:</p> <ul style="list-style-type: none">- Dimensiones de la imagen deseada.- Datos del material registrados en el catálogo.
	<p>1.3. El Diseñador diseña la imagen solicitada y la entrega al Supervisor.</p>
	<p>1.4. El Supervisor evalúa la calidad del diseño realizado.</p>
	<p>1.5. Si la calidad del diseño es satisfactoria, el Supervisor imprime la imagen diseñada.</p>
	<p>1.6. El Supervisor guarda en un disco duro una copia del material audiovisual descrito.</p>
	<p>1.7. El Supervisor guarda otra copia en un contenedor (CD o DVD) acompañada de la imagen impresa con anterioridad.</p>
	<p>1.8. El Supervisor asigna código al contenedor en el que se guardó la segunda copia del material audiovisual.</p>
	<p>1.9. El Supervisor actualiza el catálogo con el código del disco duro y del contenedor en los que se guardó el material.</p>
	<p>1.10. El Supervisor guarda el contenedor en el Archivo.</p>
	<p>1.11. Borra las copias del material existentes en</p>

CAPÍTULO 3: “Presentación de la solución propuesta”.

	los servidores de edición y de descripción.
	1.12. Informa al Coordinador de Archivología que el material ya está almacenado correctamente.
Flujo Alternativo 1	
Acción del actor	Respuesta del proceso de negocio
	1.5. Si la calidad del diseño no es satisfactoria, el Supervisor reasigna la tarea de diseño a un diseñador (se retoma el flujo normal de los eventos a partir de la acción 3)
Mejoras propuestas	

Tabla 6: Descripción textual del CUN Almacenar Material.

Diagrama de Actividades del Caso de Uso del Negocio Almacenar Material

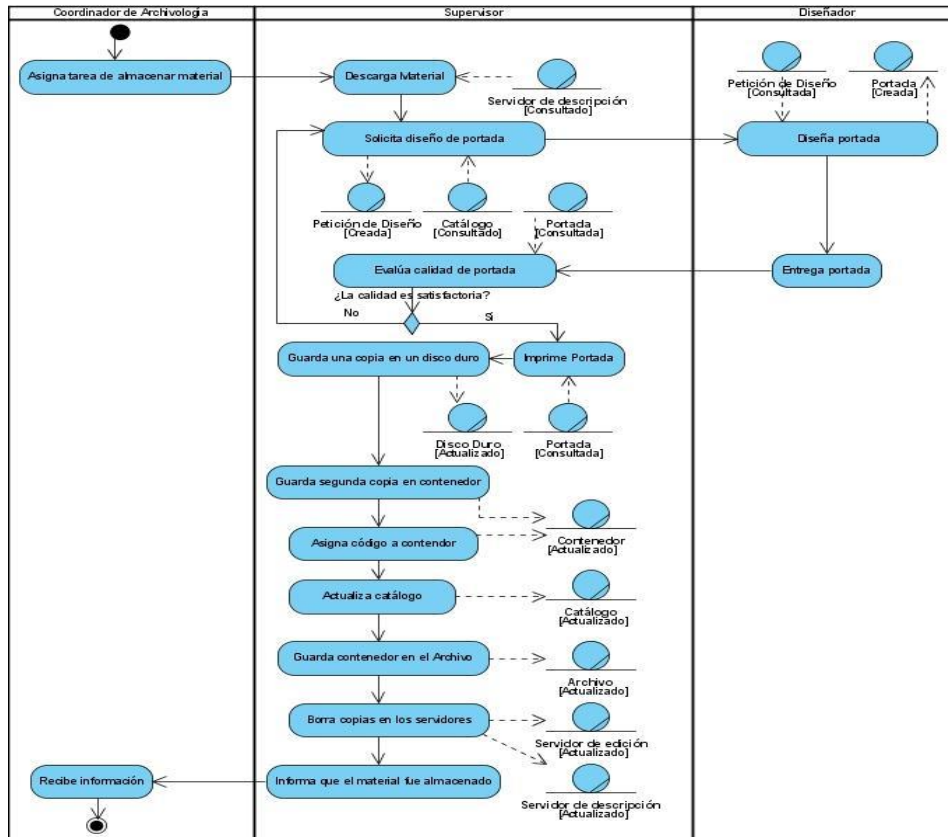


Imagen 9: Diagrama de Actividades del Caso de Uso del Negocio Almacenar Material.

3.3 Requisitos del Sistema

Según Rogers Pressman, la parte más difícil en la construcción de un software es decidir qué se va a construir. Asimismo plantea que no hay otra parte del trabajo conceptual que se torne tan dificultosa como la de establecer los requisitos técnicos detallados. Ninguna otra parte del trabajo puede perjudicar tanto el resultado final si se realiza de manera errónea.

Los requisitos son usados para establecer y mantener un acuerdo entre los usuarios, clientes y desarrolladores sobre lo que se va a construir. Sirven para definir los límites del sistema. Permite proveer una base para la estimación del costo y tiempo de desarrollo de la herramienta. En fin, los requisitos se pueden definir como la condición o capacidad que necesita cumplir un sistema o parte de él, para satisfacer las necesidades del cliente.

3.3.1 Requisitos Funcionales

RF 1. El sistema debe ser capaz de realizar la inserción de las fallas de los materiales audiovisuales que han sido digitalizados.

RF 2. El sistema debe ser capaz de listar todas las fallas correspondientes a un material audiovisual especificado.

RF 3. El sistema debe permitir a los supervisores asignar las de tareas de edición y descripción.

RF 4. El sistema debe permitir a los supervisores modificar las tareas que han sido asignadas.

RF 5. El sistema debe permitir a los editores actualizar del estado de realización de las tareas.

RF 6. El sistema debe permitir la consulta de tareas existentes.

RF 7. El sistema debe permitir a los supervisores la eliminación de una tarea seleccionada.

RF 8. El sistema debe permitir la creación de un proyecto, el mismo contendrá la información de entrada de un nuevo material al sistema y estará asociado además al registro de las fallas de dicho material.

RF 9. El sistema debe permitir la copia de un material audiovisual desde la PC local al servidor o desde el servidor a la PC local, donde será almacenado para su posterior uso.

RF 10. El sistema debe permitir la autenticación de los usuarios, identificando a los mismos y autorizando su acceso al sistema.

RF 11. El sistema debe permitir la posibilidad de describir un material audiovisual que ya haya sido digitalizado y editado. Posibilitará la entrada de datos que caractericen dicho material garantizando información del mismo.

CAPÍTULO 3: “Presentación de la solución propuesta”.

RF 12. El sistema debe ser capaz de realizar la petición de confirmación cuando el usuario desee hacer una acción irreversible del sistema.

3.3.2 Requisitos no Funcionales

Usabilidad

- **RNF 1.** Debe poder ser usado por usuarios que no tengan gran conocimiento informático, por lo que el sistema brindará un fácil acceso a sus funcionalidades.
- **RNF 2.** Se requerirá de un previo entrenamiento para que los usuarios estén listos para utilizar la aplicación.

Seguridad

- **RNF 3.** Cada usuario contará con una contraseña para acceder a la aplicación, esta será codificada y guardada en la base de datos.
- **RNF 4.** Se debe establecer a cada usuario un nivel de acceso determinado que le permitirá interactuar con el sistema según el rol que desempeñe.

Disponibilidad

- **RNF 5.** El sistema deberá estar disponible las 24 horas del día de toda la semana.
 - Tiempo medio de reparación: el sistema no debe estar fuera de servicio más de 2 horas a partir de un fallo.

Restricciones del Diseño

- **RNF 6.** Los lenguajes de programación utilizados para la implementación del Módulo de Control de Ingesta serán HTML, PHP, CSS y JavaScript.
- **RNF 7.** El sistema será implementado usando como IDE el NetBeans 6.9.
- **RNF 8.** Se utilizará para el modelado del sistema el Visual Paradigm como Herramienta CASE.

Portabilidad

- **RNF 9.** El sistema deberá ser capaz de funcionar en cualquier sistema operativo.

Hardware

CAPÍTULO 3: “Presentación de la solución propuesta”.

- **RNF 10.** En la PC cliente:
 - Procesador: Core2Duo
 - Memoria: 512 MB
 - Almacenamiento: 80 GB

- **RNF 11.** En los servidores:

Mínimo:

- Arquitectura 64 bits (x64)
- Procesador: Core2Duo
- Memoria: 1 GB

Software

- **RNF 12.** En las PC clientes:
 - Sistema Operativo: Cualquier SO
 - Navegador: Cualquier navegador
- **RNF 13.** En los servidores:
 - Sistema Operativo: OpenSuse 11.3

Sistema Gestor de Base de Datos:

- PostgreSQL 8.4 (server BD) Licencia Código Abierto. TPL.
- pgAdmin3, Licencia Código Abierto. TPL.

Aplicación Servidor Web

- Apache Server, Licencia ASF, compatible con GPL.
- Framework Symfony
- Framework Dojo

Eficiencia

- **RNF 14.** El sistema debe ser eficiente a las peticiones realizadas en cada momento, el flujo de trabajo que sigue la aplicación no permite el fallo de ninguna de las partes pues este influye de manera drástica sobre el próximo paso.

Interfaz

- **RNF 15.** La interfaz gráfica de la aplicación debe concebirse con un ambiente sencillo y de navegación fácil e intuitiva para el usuario. Es conveniente no mostrarle al usuario muchos

CAPÍTULO 3: “Presentación de la solución propuesta”.

elementos funcionales al mismo tiempo con el objetivo de que este no pierda el hilo del proceso que está realizando y le sea más lógico el uso de cada una de las funcionalidades.

Interfaces de comunicación

- **RNF 16.** El sistema requiere de comunicación a través de los protocolos TCP/IP, HTTP y FTP.

Luego de realizarse el modelo de negocio y determinarse los requisitos del software es posible modelar y proponer el sistema que se desea desarrollar.

3.4.1 Descripción de los Actores del Sistema

Un actor del sistema no es más que el rol que desempeña cualquier persona u otro sistema que interactúa con el sistema que se pretende desarrollar. Seguidamente se relacionan los actores del sistema con sus respectivas descripciones.

Actor	Descripción
Digitalizador	Es el encargado de registrar, editar, eliminar, listar y generar un reporte de las fallas de las medias, tanto de audio como de video.
Editor	Es el responsable de actualizar el estado de las tareas de edición que tenga asignada. Es quien realiza la descarga de materiales o proyectos. Es el encargado de describir los materiales.
Usuario de ingesta	Es el encargado de la subida de los materiales y proyectos al sistema. Es quien consulta las tareas existentes. Es quien realiza la autenticación de los usuarios.
Supervisor	Es el responsable de asignar, modificar, actualizar y eliminar tareas existentes. Es el encargado de generar reportes de las tareas existentes.

Tabla 7: Actores del Sistema y sus Descripciones.

3.4.2 Diagrama de Casos de Uso del Sistema

Los diagramas de casos de uso del sistema son utilizados para capturar los requisitos funcionales de este, ya que son los que representan la capacidad o condiciones que el sistema debe satisfacer. Mediante estos diagramas se pueden representar las interacciones que existen entre los actores y las funcionalidades del sistema.

CAPÍTULO 3: “Presentación de la solución propuesta”.

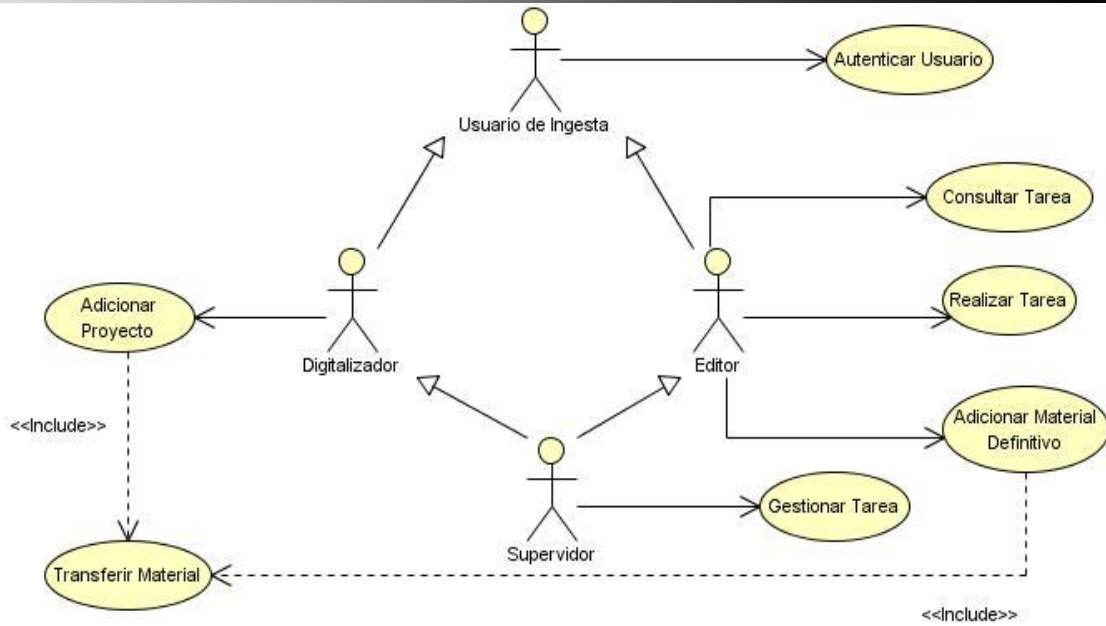


Tabla 8: Diagrama de Casos de Uso del Sistema.

3.4.3 Descripción de los de los Casos de Uso del Sistema

A continuación se muestra la descripción de 6 casos de uso del sistema de los 7 identificados. El caso de uso de Consultar Tarea se puede encontrar en los anexos del presente documento.

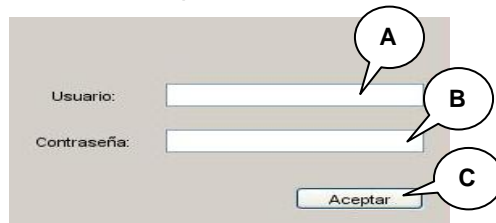
Descripción del CUS Autenticar Usuario

Caso de Uso:	Autenticar usuario
Actores:	Usuario de Ingesta, Digitalizador, Editor, Supervisor.
Resumen:	El caso de uso se inicia cuando se introduce el usuario y la contraseña para acceder en la aplicación, estos son verificados; el caso de uso finaliza mostrando la interfaz correspondiente según los permisos del usuario que se autentica.
Precondiciones:	El usuario debe existir en la Base de Datos y en el servidor LDAP y debe estar habilitado en el sistema.
Referencias	RF 10, 12.
Prioridad	Crítico
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema

CAPÍTULO 3: “Presentación de la solución propuesta”.

<p>1. El caso de uso se inicia cuando el actor escribe su nombre de usuario (A) y contraseña (B). Debe presionar el botón “Aceptar” (C) para entrar en el sistema.</p>	<p>1.1. El sistema verifica que exista conexión con la base de datos.</p> <p>1.2. El sistema verifica que el usuario exista en la Base de Datos y coincida su contraseña.</p> <p>1.3. El sistema verifica los permisos que tiene asignados y en dependencia de los permisos le permite acceder a la aplicación mostrando la interfaz correspondiente. Culmina así el caso de uso.</p>
--	---

Prototipo de Interfaz



Flujos Alternos

Acción del Actor	Respuesta del Sistema
	<p>1.2 Si no se puede establecer conexión con la Base de Datos, el sistema muestra el siguiente mensaje: “No se puede establecer conexión con los datos”, y culmina el caso de uso.</p>
	<p>1.3 Si el usuario no existe en la Base de Datos del sistema, se muestra un mensaje indicando que los datos de autenticación no son correctos y culmina el caso de uso.</p>
	<p>1.3 Si no coincide la contraseña del usuario se establece conexión con el servidor LDAP, se comparan las contraseñas del usuario y si coinciden las mismas se actualiza la contraseña de la Base de Datos del sistema. Ir a la acción 1.3 del Flujo normal de eventos.</p>

CAPÍTULO 3: “Presentación de la solución propuesta”.

	1.4 Si no se puede establecer conexión con el servidor LDAP, el sistema muestra el siguiente mensaje: “No se puede establecer conexión con el servidor LDAP”, y culmina el caso de uso.
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
	1.4 Si los datos de autenticación introducidos por el actor no son válidos, el sistema muestra el siguiente mensaje: “Usuario o contraseña incorrecta”. Culmina el caso de uso.
Poscondiciones	Se muestra la interfaz correspondiente según los permisos que posee el actor.

Tabla 9: Descripción textual del CUS Autenticar Usuario.

Descripción del CUS Adicionar Proyecto

Caso de Uso:	Adicionar Proyecto
Actores:	Digitalizador, Supervisor
Resumen:	El caso de uso se inicia cuando el digitalizador o supervisor del área de digitalización desea agregar un proyecto digitalizado al sistema.
Precondiciones:	El usuario debe estar autenticado, debe poseer permisos para subir proyectos de materiales digitalizados al sistema.
Referencias	RF 8, 9 y 12.
Prioridad	Crítico
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El caso de uso se inicia cuando el actor accede a la funcionalidad de subir proyectos.	1.1. El sistema muestra la interfaz correspondiente para introducir los datos requeridos y agregar el proyecto al sistema.
2. El actor introduce los datos necesarios para insertar un proyecto al sistema. <ul style="list-style-type: none"> Nombre (Obligatorio) (B) 	

CAPÍTULO 3: “Presentación de la solución propuesta”.

<ul style="list-style-type: none"> • Restricción (Obligatorio) (C) • Cruce de cassettes (Obligatorio) (D) • Código de cassette (Obligatorio) (E) • Formato (Obligatorio) (Y) • Nivel de video inicial (Obligatorio) (F) • Nivel de video final(Obligatorio) (G) • Nivel de pedestal inicial (Obligatorio) (H) • Nivel de pedestal final (Obligatorio) (I) • Nivel de croma inicial (Obligatorio) (J) • Nivel de croma final (Obligatorio) (K) • Nivel de color inicial (Obligatorio) (L) • Nivel de color final (Obligatorio) (M) • Nivel de audio inicial (Obligatorio) (N) • Nivel de audio final (Obligatorio) (O) • Seleccionar archivo (Obligatorio) (A) • Seleccionar las fallas (Obligatorio) (P) • Descripción (Obligatorio) (Q). 	
<p>3. El usuario presiona el botón “Fallas correspondientes al proyecto” (P)</p>	<p>3.1. El sistema muestra una interfaz para seleccionar las fallas que contiene el proyecto.</p>
<p>4. El usuario introduce los datos de las fallas.</p> <ul style="list-style-type: none"> • Nombre de falla (Obligatorio) (R). • Intensidad (Obligatorio) (S). • Duración (Obligatorio) (T). • Tiempo de inicio (Obligatorio) (U). • Tiempo de fin (Obligatorio) (V). 	
<p>5. El usuario presiona el botón “Aceptar” (W).</p>	<p>5.1. El sistema crea una lista de las fallas que corresponden al proyecto.</p>
<p>6. El usuario presiona el botón “Guardar” (X)</p>	<p>6.1. El sistema comprueba que los datos introducidos estén completos y realiza la transferencia del proyecto hacia el directorio temporal donde será almacenado para posteriormente acceder a él. Ir al CUS</p>

CAPÍTULO 3: "Presentación de la solución propuesta".

Transferir Material. Culmina el caso de uso.

Prototipo de Interfaz

The screenshot shows the 'Adicionar Proyecto' (Add Project) interface. At the top, there are two tabs: 'Adicionar Proyecto' (selected) and 'Gestionar Proyecto'. Below the tabs, there are several input fields and dropdown menus:

- Nombre:** A text input field with callout B.
- Restricción:** A dropdown menu with '-Seleccione-' selected, callout C.
- Cruce de Cassettes:** A text input field with callout D.
- Formato:** A dropdown menu with '-Seleccione-' selected, callout Y.
- Código del cassette:** A text input field with callout E.

Below these fields is a section titled 'Niveles de los Parámetros de Imagen y Sonido' (Image and Sound Parameters Levels). It contains two columns of sliders, each with a label and a numerical value (0,0):

- Left column: Nivel de Video (callout F), Nivel de Pedestal (callout H), Nivel de Croma (callout J), Nivel de Color (callout L), Nivel de Audio (callout N).
- Right column: (callout G), (callout I), (callout K), (callout M), (callout O).

Below the sliders, there is a text input field with callout A, an 'Examinar...' button, a link 'Subir otro archivo', and a text box 'Fallas correspondientes al proyecto' with callout P. At the bottom, there is an 'Observaciones' section with a large text area and callout Q, and a 'Guardar' button with callout X.

The screenshot shows a dialog box titled 'Seleccionar Fallas del Proyecto' (Select Project Failures). It contains a section titled 'Niveles de los Parámetros de Imagen y Sonido' (Image and Sound Parameters Levels) with the following fields:

- Nombre de falla:** A dropdown menu with '-Seleccione-' selected, callout R.
- Intensidad:** A dropdown menu with '-Seleccione-' selected, callout S.
- Duración:** A dropdown menu with '-Seleccione-' selected, callout T.
- Tiempo de Inicio:** Three spinners, each with '0', callout U.
- Tiempo de Fin:** Three spinners, each with '0', callout V.

At the bottom of the dialog is an 'Aceptar' button with callout W.

Poscondiciones

Se inserta el proyecto de digitalización en el sistema quedando disponible para su edición.

CAPÍTULO 3: “Presentación de la solución propuesta”.

Tabla 10: Descripción textual del CUS Adicionar Proyecto.

Descripción del CUS Transferir Material

Caso de Uso:	Transferir Material
Actores:	CUS Adicionar Proyecto, CUS Adicionar Material Definitivo.
Resumen:	El caso de uso se inicia cuando es necesaria la ejecución de transferencia de ficheros hacia o desde el servidor.
Precondiciones:	Que el usuario esté autenticado, tenga permiso para adicionar o descargar materiales.
Referencias	RF 9.
Prioridad	Crítico
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
	1.1 El caso de uso de inicia cuando el sistema verifica la existencia de espacio suficiente para el copiado del fichero en la dirección de destino correspondiente.
	1.2 El sistema ejecuta la transferencia de fichero de acuerdo a las direcciones origen y destino establecidas por el usuario y/o por el sistema, mostrando el porcentaje de realización de la transferencia hasta el fin de ella, concluyendo así el caso de uso.
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
	1.1 En caso de que no exista espacio suficiente para realizar la transferencia, el sistema muestra un error indicándolo, terminando así el caso de uso.
	2.1 En caso de ocurrir algún error de conexión durante la transferencia del fichero, el sistema muestra un mensaje de error indicándolo y activa la contingencia de

CAPÍTULO 3: “Presentación de la solución propuesta”.

	fallas para este caso.
	<p>3.1 Se almacena la copia del fragmento de fichero que se transfiere temporalmente (1 hora).</p> <p>a) Si dentro de ese periodo de tiempo no se ha restablecido la conexión se elimina el fragmento de fichero y concluye el caso de uso.</p> <p>b) Si se restablece la conexión en ese periodo de tiempo se reanuda la transferencia del fichero y concluye el caso de uso cuando finalice la misma.</p>
Poscondiciones	Se realiza la transferencia de ficheros desde y hacia el servidor.

Tabla 11: Descripción textual del CUS Transferir Material.

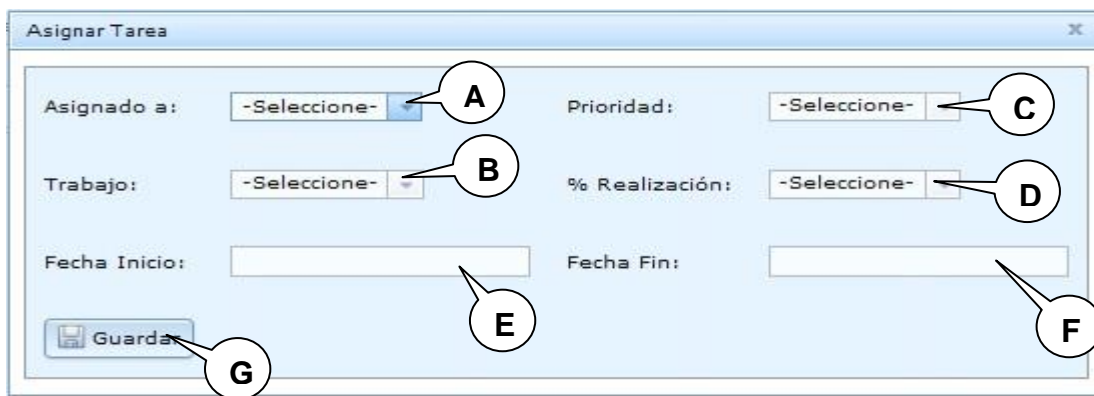
Descripción del CUS Gestionar Tarea

Caso de Uso:	Gestionar Tarea	
Actores:	Supervisor	
Resumen:	El caso de uso se inicia cuando el supervisor del área de digitalización necesita insertar, modificar o eliminar las tareas de edición del sistema.	
Precondiciones:	El usuario debe estar autenticado, debe poseer permisos para gestionar tareas de edición.	
Referencias	RF 3, 4, 7 y 12.	
Prioridad	Crítico	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Sistema	
1. El caso de uso se inicia cuando el usuario selecciona la opción de “Tareas”. Puede seleccionar una de las siguientes tres opciones: a) Asignar tareas, ir a la sección “asignar tareas”.		

CAPÍTULO 3: “Presentación de la solución propuesta”.

b) Editar tarea, ir a la sección “editar tarea”. c) Eliminar tarea, ir a la sección “Eliminar tarea”.	
Sección “Asignar tarea”	
Acción del Actor	Respuesta del Sistema
2. El usuario introduce los datos necesarios para la creación de una tarea. a) Asignado a (Obligatorio) (A) b) Trabajo (Obligatorio) (B) c) Prioridad (Obligatorio) (C) d) Realización (Obligatorio) (D) e) Fecha Inicio (Obligatorio) (E) f) Fecha Fin (Obligatorio) (F)	1.1. El sistema muestra la interfaz para insertar los datos de la tarea.
3. El usuario presiona el botón “Guardar” (G).	3.1. El sistema muestra un mensaje indicando que la tarea ha sido creada satisfactoriamente. Culmina el caso de uso.

Prototipo interfaz

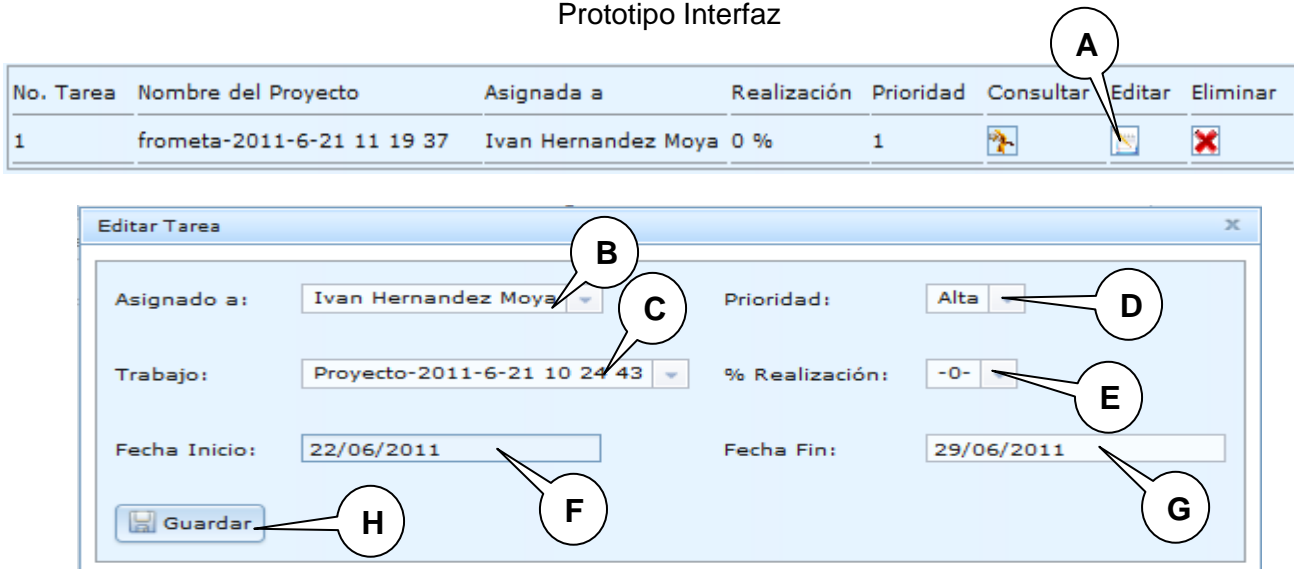


Sección “Editar tarea”	
Acción del Actor	Respuesta del Sistema
	1.1. El sistema muestra la lista de tareas que

CAPÍTULO 3: “Presentación de la solución propuesta”.

	no han sido realizadas.
2. El usuario selecciona la opción “Editar” (A) de la tarea que desea modificar.	2.1. El sistema carga los datos de la tarea seleccionada en una interfaz que permite su modificación.
3. El usuario modifica los datos de la tarea. a) Asignado a (Obligatorio) (B) b) Trabajo (Obligatorio) (C) c) Prioridad (Obligatorio) (D) d) Realización (Obligatorio) (E) e) Fecha Inicio (Obligatorio) (F) f) Fecha Fin (Obligatorio) (G)	
4. El usuario presiona el botón “Guardar” (H).	4.1. El sistema muestra un mensaje indicando que la modificación de la tarea ha sido realizada satisfactoriamente. Culmina el caso de uso.

Prototipo Interfaz



Sección “Eliminar tarea”

Acción del Actor

Respuesta del Sistema

1.1. El sistema muestra el listado de las tareas

CAPÍTULO 3: “Presentación de la solución propuesta”.

	que no han sido resueltas.																
2. El usuario selecciona la opción “Eliminar” (A) de la tarea que desea eliminar.	2.1. El sistema muestra un mensaje de confirmación de la acción.																
3. El usuario confirma la acción.	3.1. La tarea es eliminada del sistema, se muestra un mensaje indicándolo. Culmina el caso de uso.																
Flujo Alternativo																	
Acción del Actor	Respuesta del Sistema																
3. El usuario no confirma a acción.	3.1. La tarea no es eliminada del sistema. Culmina el caso de uso.																
<i>Prototipo de Interfaz</i>																	
<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>No. Tarea</th> <th>Nombre del Proyecto</th> <th>Asignada a</th> <th>Realización</th> <th>Prioridad</th> <th>Consultar</th> <th>Editar</th> <th>Eliminar</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>frometa-2011-6-21 11 19 37</td> <td>Ivan Hernandez Moya</td> <td>0 %</td> <td>1</td> <td></td> <td></td> <td> A</td> </tr> </tbody> </table>		No. Tarea	Nombre del Proyecto	Asignada a	Realización	Prioridad	Consultar	Editar	Eliminar	1	frometa-2011-6-21 11 19 37	Ivan Hernandez Moya	0 %	1			A
No. Tarea	Nombre del Proyecto	Asignada a	Realización	Prioridad	Consultar	Editar	Eliminar										
1	frometa-2011-6-21 11 19 37	Ivan Hernandez Moya	0 %	1			A										
Poscondiciones	Se insertan, modifican o eliminan los datos de las tareas presentes en el sistema.																

Tabla 12: Descripción textual del CUS Gestionar Tarea.

Descripción del CUS Realizar Tareas

Caso de Uso:	Realizar Tarea
Actores:	Editor, Supervisor
Resumen:	El caso de uso se inicia cuando el editor o supervisor del área de digitalización realizará una de las tareas de edición que le han sido asignadas.
Precondiciones:	El usuario debe estar autenticado, debe poseer permisos para realizar tareas de edición.
Referencias	RF 5.
Prioridad	Crítico
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
	1. El caso de uso se inicia cuando sistema

CAPÍTULO 3: “Presentación de la solución propuesta”.

	muestra una interfaz con la opción de descargar el material correspondiente a la tarea seleccionada y que muestra las fallas de audio y video que le han sido notificadas al mismo.
2. El usuario selecciona la opción “Descargar” (A).	2.1 El sistema muestra solicita la dirección de destino del proyecto.
3. El usuario introduce la dirección de destino donde se copiará el proyecto a editar.	3.1 Ir al CUS Transferir Material. Culmina el caso de uso.
4. El usuario selecciona actualizar el porciento de realización de la tarea (Opcional) (B).	4.1 El sistema habilita la selección del nuevo por ciento.
5. El usuario selecciona el porciento de realización de la tarea. (Obligatorio) (C).	
6. El usuario presiona el botón “Actualizar” (D).	6.1 El sistema actualiza el porciento de la tarea.
7. El usuario selecciona actualizar las observaciones del proyecto. (Opcional) (E).	7.1 El sistema habilita el área de texto para la nueva observación.
8. El usuario introduce la nueva observación (Obligatorio) (F).	
9. El usuario presiona el botón “Actualizar” (D).	
10. El usuario selecciona la opción “Subir” (G).	10.1 El sistema muestra una interfaz para subir los archivos actualizados pertenecientes al proyecto.
11. El usuario selecciona subir otro archivo (Opcional) (H).	11.1 El sistema crea otro botón para buscar otro archivo.
12. El usuario selecciona el archivo (Obligatorio) (I).	
13. El usuario presiona el botón Enviar (Obligatorio) (J).	

Prototipo de Interfaz

CAPÍTULO 3: “Presentación de la solución propuesta”.

Consultar Tarea

Asignado a: documentalista Prioridad: 1

Fecha Realización.: 2011-06-20 % Realización: 0%

Fecha Inicio: 2011-06-20 Fecha Fin: 2011-06-29

Nombre del proyecto: proyecto UCI-2011-6-20 09 40 40

Nombre de la falla:	Intensidad:	Duración:	Tiempo de inicio:	Tiempo de fin:
Audio comprimido	Ligeramente molesto	Durante varios periodos cortos	0:9:0	0:12:0
Fuera de convergencia	Ligeramente molesto	Durante varios periodos prolongados	0:6:0	0:12:0

Observaciones: Esta tarea está en un 0%...

Actualizar Descargar Subir

Subier Archivos

Archivos a Subir: Examinar...

Subir otro archivo

Enviar

Poscondiciones

Se actualiza el estado de la tarea asignada y se muestra al editor las fallas que posee el proyecto que va a editar.

Tabla 13: Descripción textual del CUS Realizar Tareas.

Descripción del CUS Adicionar Material Definitivo

Caso de Uso:	Adicionar Material Definitivo
Actores:	Editor, Supervisor
Resumen:	El caso de uso se inicia cuando el editor o supervisor del área de digitalización necesita agregar un material editado al sistema.
Precondiciones:	El usuario debe estar autenticado, debe poseer permisos para insertar materiales definitivos en el sistema.
Referencias	RF 11 y 9.

CAPÍTULO 3: “Presentación de la solución propuesta”.

Prioridad	Crítico	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Sistema	
1. El caso de uso se inicia cuando el usuario selecciona la opción de “Archivar Material” al servidor.	1.1. El sistema muestra la interfaz para la inserción de datos necesarios cuando se adiciona un material definitivo.	
2. El usuario introduce los datos necesarios: <ul style="list-style-type: none"> a) Soporte (Obligatorio) (A) b) Procedencia (Obligatorio) (B) c) Restricciones (Obligatorio) (C) d) Cargar Material (Obligatorio) (D) e) Descripción (Obligatorio) (E) 		
3. El usuario presiona el botón “Guardar”. (F)	3.1. Ir al CUS “Transferir Material”. Culmina el caso de uso.	

Prototipo de Interfaz

Archivar Material Maestro

Soporte: Procedencia:

Restricciones:

Cargar Material:

Descripción

Poscondiciones	Se adiciona un nuevo material al sistema, listo para ser procesado.
----------------	---

Tabla 14: Descripción textual del CUS Adicionar Material Definitivo.

3.5 Conclusiones

Durante este capítulo han quedado expuestos los artefactos principales del flujo de trabajo: Levantamiento de Requisitos. A través del Modelo de Negocio, se pudo conocer y llegar a un mejor entendimiento de los procesos existentes y más potenciales a automatizar, en la solución del Subsistema de Control de Ingesta. Se identificaron los requisitos funcionales y los no funcionales pertenecientes al sistema propuesto, con la finalidad de dar a conocer las necesidades que debe satisfacer el mismo. Se hizo un reconocimiento de los actores y trabajadores del negocio, así como la identificación, representación y descripción de los casos de uso. Se definieron, representaron y describieron los casos de uso del sistema con el fin de proporcionar un mejor acercamiento y comprensión de sus funcionalidades.

Capítulo 4: “Construcción de la solución propuesta”.

4.1 Introducción

El actual capítulo tiene como propósito mostrar una descripción de la solución que se propone para la presente investigación, basándose en la en los artefactos generados durante los flujos de trabajo de Análisis y Diseño, Implementación y Prueba. Se definirá la estructura del diagrama de diseño y la implementación de la aplicación a desarrollar, además de la representación de los diagramas de clases de diseño, el diagrama de clases persistentes, el diagrama de componentes y el modelo de despliegue.

4.2 Arquitectura del Software

En el desarrollo de un software es muy común encontrar diversos puntos de vistas entre los equipos de trabajo que asumirán la construcción del sistema, en este caso la arquitectura es el instrumento encargado para unificar de manera sencilla estas diferencias. Es la primera etapa del proceso de Ingeniería de Software que consiste en producir un modelo o representación técnica del software que se va a construir. Permite identificar los elementos más importantes de un sistema así como sus relaciones, proporcionando una visión global del sistema.

La Arquitectura de Software es de mucha importancia debido a que facilitan la comunicación entre todas las partes interesadas en el desarrollo de un sistema. Destaca decisiones tempranas de diseño que tendrán un profundo impacto en todo el trabajo de Ingeniería del software. Constituye un modelo relativamente pequeño e intelectualmente comprensible de cómo está estructurado el sistema y cómo trabajan sus componentes.

Debido a las diversas Metodologías de Desarrollo de Software que existen en la actualidad, se hace complicado definir una visión unificada sobre el diseño arquitectónico. Claro está que no importa el carácter de la metodología usada, siempre existirá una fase donde se implantará la arquitectura base. En el presente trabajo de diploma se utiliza RUP como Metodología de Desarrollo para la conceptualización y construcción del sistema, por lo que la arquitectura se establece en la fase de elaboración.

4.3 Patrones

G. Booch plantea que “Los patrones conducen a arquitecturas más pequeñas, más simples y más comprensibles.”

Como tal un patrón es un modelo que se puede seguir para realizar algo. Los mismos están basados en experiencias acumuladas por los desarrolladores, con el objetivo de describir las soluciones más factibles a problemas comunes.

Los patrones permiten atacar a problemas concurrentes que ocurren en situaciones específicas y dan una solución. Documentan experiencias de diseños existentes y bien probados. Permiten identificar y especificar abstracciones de alto nivel. Proveen un vocabulario común y comprensible y son una forma de documentar la Arquitectura de Software.

Los patrones se pueden agrupar en diferentes categorías según su nivel de abstracción: **(25)**

Patrones de Análisis: Reflejan estructuras conceptuales de procesos de negocios más que implementaciones de software.

Patrones arquitectónicos: Describen los principios fundamentales de la arquitectura de un sistema de software. Identifican los subsistemas, define sus responsabilidades y establece las reglas y guías para organizar las relaciones entre ellos.

Patrones de diseño: Los patrones de diseño proveen un esquema para refinar los subsistemas y las relaciones entre los componentes de un sistema. Describen una estructura recurrente de comunicación entre componentes para resolver un problema general de diseño dentro de un contexto particular.

Expresiones idiomáticas: Los modismos o expresiones idiomáticas son el nivel más bajo de abstracción en un sistema de patrones. Describen como implementar una parte particular de diseño al utilizar un lenguaje de programación determinado.

4.3.1 Patrón de Arquitectura

Entre los patrones de arquitectura más conocidos se encuentra el Modelo Vista Controlador (MVC) que se puede observar con mucha frecuencia en aplicaciones Web. Este se separa en tres componentes distintos; el Modelo que es quien representa la información con la que trabaja la

CAPÍTULO 4: “Construcción de la solución propuesta”.

aplicación, es decir su lógica de negocio; la Vista que es la que transforma el modelo en una página Web que permite al usuario interactuar con ella y el Controlador que se encarga de procesar las interacciones del usuario y realiza los cambios apropiados en la vista o en el modelo.

Para la construcción del sistema propuesto se decidió utilizar este patrón debido a que el sistema que se obtendrá como resultado es una aplicación Web desarrollada con el Framework de desarrollo Symfony. El mismo está basado en la arquitectura MVC, por lo que se determinó que este patrón es el más apropiada para el desarrollo de la aplicación de la presente investigación.

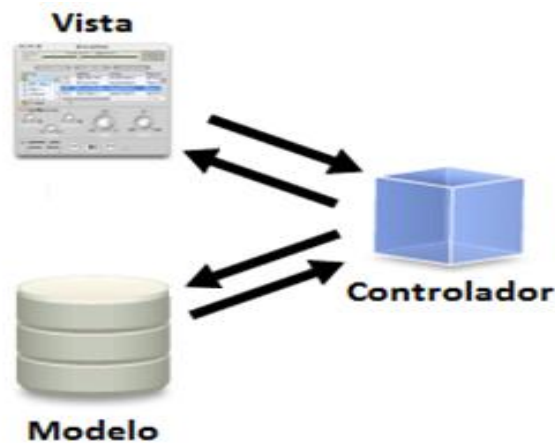


Imagen 10: Modelo Vista Controlador.

Esta arquitectura separa la lógica del negocio (el modelo) y la presentación (la vista) garantizando un mantenimiento más sencillo de la aplicación. El controlador se encarga de aislar al modelo y a la vista de los detalles del protocolo usado para las peticiones (HTTP, consola de comandos, e mail, etc.). El modelo se encarga de la abstracción de la lógica relacionada con los datos, haciendo que la vista y las acciones sean independientes de, por ejemplo, el tipo de gestor de base de datos utilizado por la aplicación.

4.3.2 Patrones de Diseño

“Un patrón de diseño es:

- Una solución estándar para un problema común de programación.
- Una técnica para flexibilizar el código haciéndolo satisfacer ciertos criterios.
- Un proyecto o estructura de implementación que logra una finalidad determinada.
- Un lenguaje de programación de alto nivel.

CAPÍTULO 4: “Construcción de la solución propuesta”.

- Una manera más práctica de describir ciertos aspectos de la organización de un programa.
- Conexiones entre componentes de programas.
- La forma de un diagrama de objetos o de un modelo de objetos.” (26)

Para la asignación de responsabilidades a objetos determinados se propone usar los patrones de diseño GRASP pero esta vez, vinculados con Symfony. Los patrones básicos de asignación de responsabilidades de GRASP son:

- *Experto*: Este es uno de los más usados debido a que hace que la clase que cuenta con la información necesaria para cumplir con una tarea sea la encargada de realizar la misma. Doctrine es una de las librerías externas que usa Symfony para realizar su capa de abstracción en el modelo, encapsulando toda la lógica de los datos y son generadas las clases con todas las funcionalidades comunes de las entidades.
- *Creador*: Este se ocupa de la creación de instancias de una clase, su tarea consiste en hallar un creador el cual se conectará con el objeto generado. En la clase Actions de Symfony es donde se crean los objetos de las clases que representan las entidades, evidenciando de este modo que la clase Actions es creador de dichas entidades.
- *Alta Cohesión*: Symfony permite asignar responsabilidades con una alta cohesión, por ejemplo la clase Actions tiene la responsabilidad de definir las acciones para las plantillas y colabora con otras para realizar diferentes operaciones, instanciar objetos y acceder a las propiedades, o sea, está formada por diferentes funcionalidades que se encuentran relacionadas estrechamente proporcionando que el software sea flexible frente a grandes cambios.
- *Bajo Acoplamiento*: Tiene como objetivo establecer una escasa dependencia entre clases, en el caso de Symfony, la clase Actions solo hereda de la sfActions logrando un bajo acoplamiento.
- *Controlador*: Define quien deberá encargarse de atender una petición del sistema. En Symfony todas las peticiones son manejadas por un controlador frontal (sfActions), que es el único punto de entrada de toda la aplicación. Cuando el controlador frontal recibe una petición, utiliza el sistema de enrutamiento para asociar el nombre de una acción y el nombre de un módulo con el url entrado por el usuario.

Symfony implementa algunos de los patrones GOF, algunos de estos son:

- En la categoría de creacionales:
 - *Singleton (Instancia única)*: Garantiza la existencia de una única instancia para una clase y la creación de un mecanismo de acceso global a dicha instancia. En el

CAPÍTULO 4: “Construcción de la solución propuesta”.

controlador frontal hay una llamada a `sfContext::getInstance()`. En una acción, el método `getContext()`, es un objeto muy útil que guarda una referencia a todos los objetos del núcleo de Symfony.

- *Abstract Factory (Fábrica abstracta)*: Permite trabajar con objetos de distintas familias de manera que las familias no se mezclen entre sí y haciendo transparente el tipo de familia concreta que se esté usando. Cuando el Framework necesita por ejemplo crear un nuevo objeto para una petición, busca en la definición de la factoría el nombre de la clase que se debe utilizar para esta tarea.
- En la categoría de estructurales:
- *Decorator (Envoltorio)*: Añade funcionalidad a una clase, dinámicamente. El archivo `layout.php`, que también se denomina plantilla global, almacena el código HTML que es común a todas las páginas de la aplicación, para no tener que repetirlo en cada página. El contenido de la plantilla se integra en el layout, o si se mira desde el otro punto de vista, el layout decora la plantilla. Este comportamiento es una implementación del patrón de diseño Decorator.
 - *Composite (Objeto compuesto)*: Permite tratar objetos compuestos como si de uno simple se tratase. Sirve para construir objetos complejos a partir de otros más simples y similares entre sí, gracias a la composición recursiva y a una estructura en forma de árbol. Esto simplifica el tratamiento de los objetos creados, ya que al poseer todos ellos una interfaz común, se tratan todos de la misma manera.

4.4 Diagrama 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 de una aproximación del sistema que se desea implementar, contribuyendo así a una mayor calidad del producto final. A continuación se muestra solo el diagrama de clases del diseño correspondiente a el caso de uso del sistema Gestionar Tarea, para lograr una mejor organización del documento, los restantes estarán expuestos en los anexos del trabajo de diploma.

CAPÍTULO 4: “Construcción de la solución propuesta”.

4.4.1 Diagrama de Clase del Diseño Gestionar Tarea.

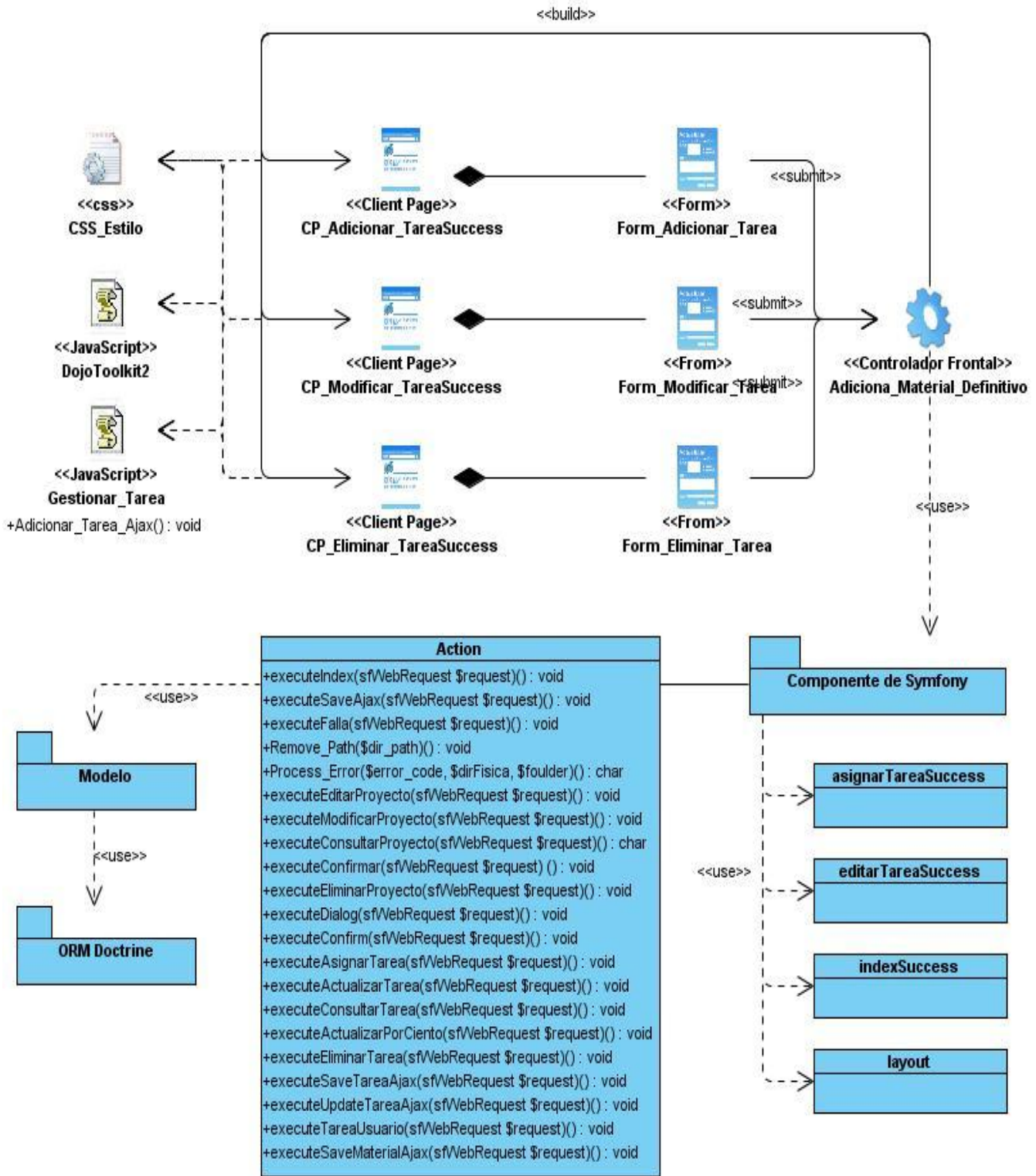


Imagen 11: Diagrama de Clases del Diseño: Gestionar Tarea.

CAPÍTULO 4: “Construcción de la solución propuesta”.

4.5 Principios de Diseño

Para el diseño gráfico de la aplicación se determinó seguir los principios que se describen a continuación. Los mismos están enfocados a lograr una interfaz efectiva, visualmente comprensiva y de fácil uso. Lo más importante es comprender que el usuario quiere hacer su trabajo de la forma más sencilla y rápida posible, y la aplicación no es más que la herramienta que va a ayudarles a lograrlo, por lo que cuanto menos estorbe la aplicación al usuario mejor. Según Alan Cooper, en su libro About Face 2.0: “No importa lo genial que sea tu interfaz, menos es más siempre”.

Unas de las leyes básicas y más conocidas en el diseño de interfaz del usuario es “La Ley de Fitt”, la misma expresa que cuanto más grande y más cercano al puntero del ratón es un objeto, más sencillo es hacer click sobre él. De aquí la importancia de hacer más grandes y más distinguibles los controles más utilizados en la aplicación. Utilizar los bordes y esquinas de las pantallas para hacer que los controles sean virtualmente infinitos. Nunca colocar los controles a un pixel de distancia del borde de la pantalla o de una esquina.

Es de suma importancia lograr que el usuario se mantenga concentrado en la tarea que está realizando, por lo que se debe minimizar la cantidad de distracción y de interferencias por parte de la aplicación.

Lograr reunir un capital humano altamente calificado es más costoso que comprar ordenadores con una mayor potencia, por tales motivos el diseño debe ir encaminado a facilitarle el trabajo al usuario, sin importar el cúmulo de cálculos adicionales que deba hacer el microprocesador.

4.6 Diseño de Base de Datos

Una de las tareas más importantes a la hora de construir un software, que maneje gran cantidad de información que debe ser almacenada, es la base de datos. Para el diseño de la misma se utiliza el modelo de datos. Claro está que una base de datos bien diseñada permite obtener acceso a la información exacta y actualizada, además de brindar notables ventajas como la integridad de los datos, mayor seguridad de los mismos, datos más documentados, acceso a datos más eficientes y menor espacio de almacenamiento. Para obtener una representación física de la base de datos se utiliza el modelo de datos, que se muestra a continuación.

CAPÍTULO 4: “Construcción de la solución propuesta”.

4.6.2 Modelo de Datos (Modelo Físico de Datos)

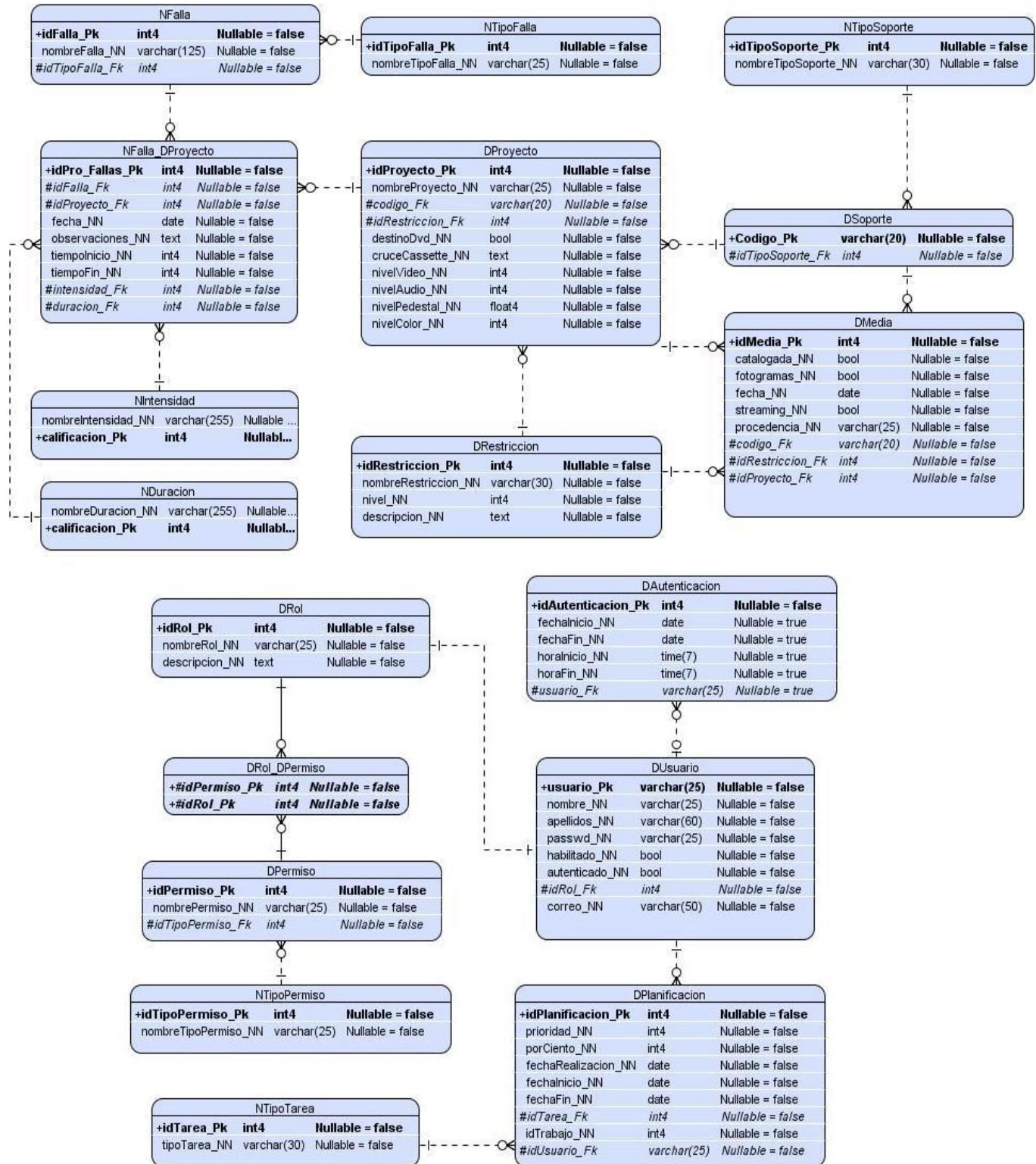


Imagen 12: Modelo de Datos.

CAPÍTULO 4: “Construcción de la solución propuesta”.

4.7 Modelo de Despliegue

El modelo de despliegue muestra las relaciones físicas entre los componentes de software y hardware en el sistema final. El mismo está compuesto por el diagrama de despliegue que se representa a continuación, en el que pueden observarse los tres nodos que componen la distribución física de la aplicación. Primeramente, el nodo “PC Cliente”; que representa a los ordenadores desde donde se interactúa con la aplicación cliente del subsistema, que es la que posee todas las interfaces gráficas de comunicación con el usuario; establece una conexión con el nodo en el que se encuentran el “Servidor de Medias” y “Servidor Web” mediante el protocolo HTTP. El “Servidor Web”, a su vez, se comunica mediante TCP/IP con el Nodo “Servidor de Base de Datos”, en el cual se encuentra almacenada toda la información perteneciente al sistema.

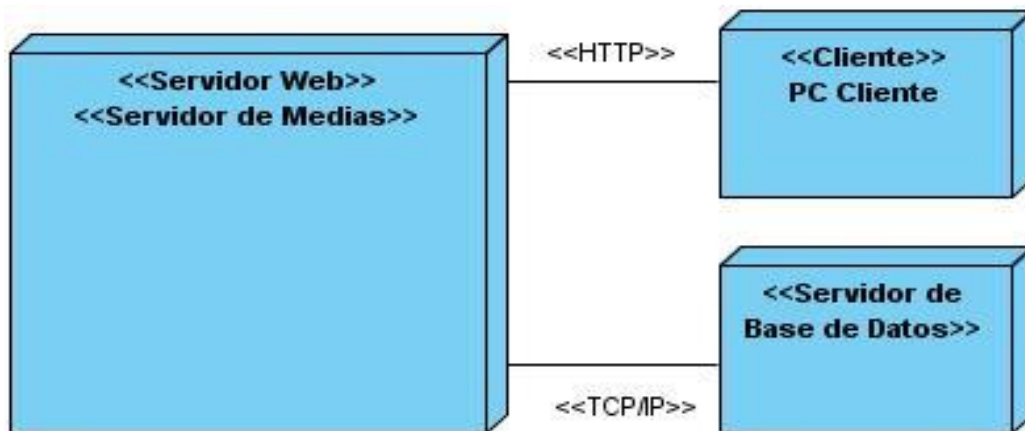


Imagen 13: Diagrama de Despliegue.

4.8 Modelo de Implementación

El modelo de implementación está formado por un conjunto de subsistemas y componentes que constituyen la composición física del sistema que se implementa. Los diagramas de componentes describen los elementos físicos del sistema y sus relaciones. Éstos pueden ser simples archivos, paquetes, bibliotecas cargadas, etc. Como tal, los componentes representan todos los tipos de elementos de software que entran en la fabricación de aplicaciones informáticas. A continuación se muestra el correspondiente a la aplicación que se desarrolla.

CAPÍTULO 4: “Construcción de la solución propuesta”.

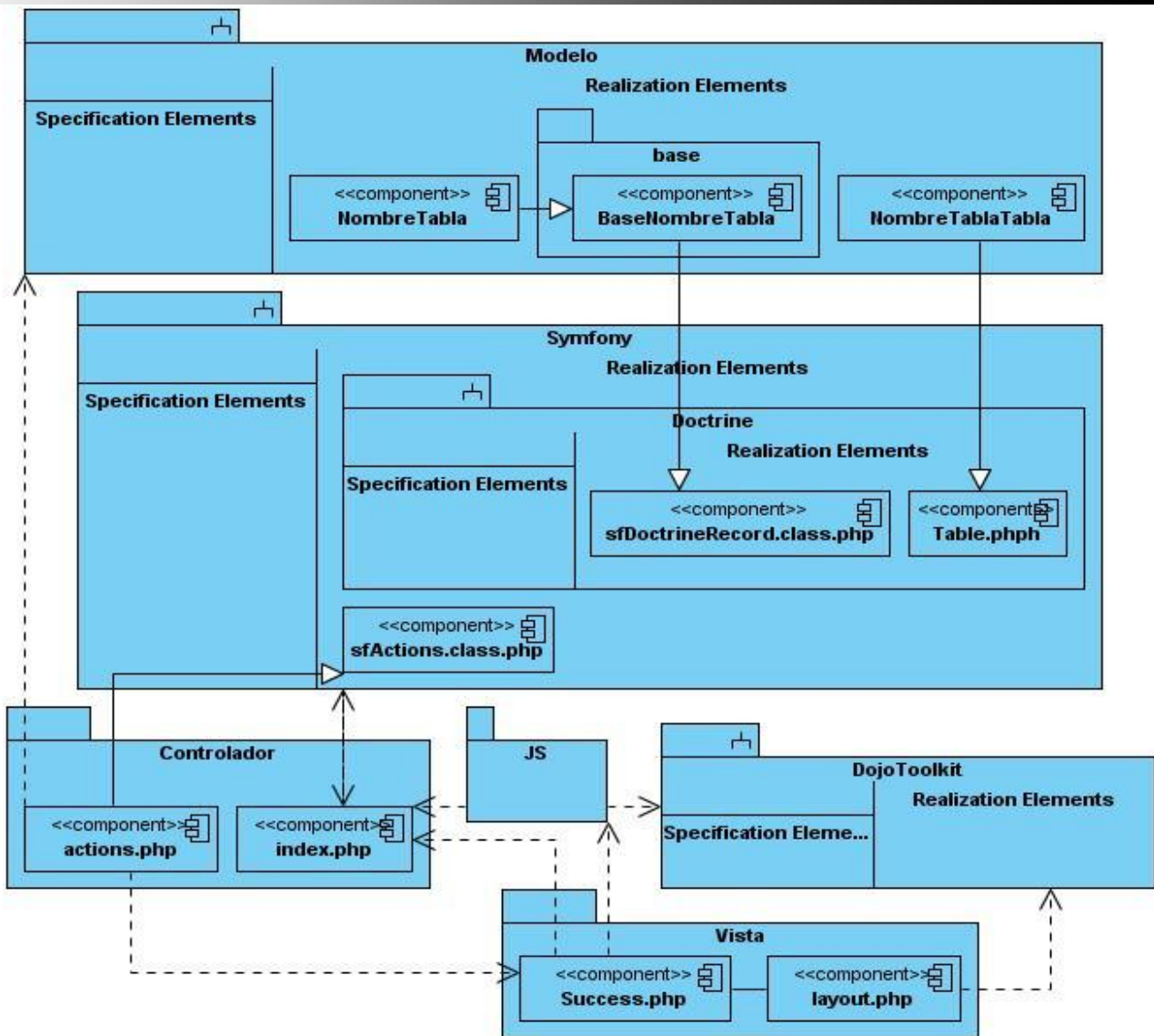


Imagen 14: Diagrama de Componentes.

4.9 Pruebas del Sistema Propuesto

Dado a que el Subsistema de Control de Ingesta es un software producto de un proceso ingenieril, del cual se conocen las funcionalidades específicas para las que fue diseñado, se pueden llevar a cabo pruebas que demuestren que cada función es completamente operativa. Este enfoque se refiere a las pruebas de caja negra que se llevan a cabo sobre la interfaz de la aplicación. Por esta razón los casos de prueba pretenden demostrar que las funcionalidades del software son operativas, que la entrada se acepta de forma adecuada y que la salida es correcta. Estas pruebas examinan algunos aspectos del sistema sin tener mucho en cuenta la estructura interna del sistema.

CAPÍTULO 4: “Construcción de la solución propuesta”.

Objetivo

El objetivo de realizar este tipo de prueba al sistema, es detectar el incorrecto o incompleto funcionamiento del mismo, así como los errores de interfaz, errores de inicialización y terminación.

Alcance

El proceso de pruebas de caja negra se va a centrar principalmente en los requisitos funcionales del software para verificar el comportamiento de la interfaz gráfica, su interacción con el usuario y la calidad funcional.

Casos de prueba

Para verificar que se cumplieron los requisitos funcionales, se realizan las pruebas a los casos de uso más críticos. Estas se hacen con el fin de evaluar la interacción del usuario con el sistema, dependiendo de las funcionalidades a las que tenga acceso el usuario y además se verifica que cada una de estas se corresponde con las descripciones de los casos de uso del sistema realizado.

4.9.1 Caso de prueba para el caso de uso “Adicionar Material Definitivo”

Nombre de la sección	Escenarios de la sección	Descripción de la funcionalidad
SC 1: Adicionar Material Definitivo	EC 1.1: Adicionar un material definitivo.	El sistema muestra un formulario para entrar los datos pertenecientes al material que se desea guardar el servidor, el usuario entra los datos correspondientes y los envía; estos se adicionan satisfactoriamente.
	EC 1.2: Tratamiento de errores	El sistema muestra un formulario para entrar los datos asociados al almacenamiento. Una vez enviados por el usuario el sistema le muestra un mensaje informando que estos no son correctos.

CAPÍTULO 4: “Construcción de la solución propuesta”.

Tabla 15: Secciones a probar en el caso de uso.

Descripción de variables.

No	Nombre de campo	Clasificación	Valor Nulo	Descripción
1	Soporte	Combo Box	No	Se tiene que seleccionar un soporte.
2	Procedencia	Combo Box	No	Se tiene que seleccionar una procedencia.
3	Restricción	Combo Box	No	Se tiene que seleccionar una restricción.
4	Ruta	Campo de texto	No	Este campo hace referencia a la ruta donde se encuentra el material audiovisual que se desea guardar.
5	Examinar	Botón	No	Permite cargar los datos introducidos por el usuario.
6	Descripción	Área de Texto	No	Se debe escribir una breve descripción referente al material audiovisual que se guardará en el servidor y que no contenga más de 255 caracteres.

Tabla 16: Descripción de variables.

Escenario	V 1 So- porte	V 2 Proce- dencia	V 3 Restric- ción	V 4 Ruta	V 6 Descri- pción	Respuesta del Sistema	Resultado de la Prueba
EC 1.1: Adicionar un material definitivo.	V	V	V	V	V	Se copia en el servidor de medias el material audiovisual y se guardan los datos correspondientes al mismo en la base de datos.	Satisfactoria.
EC 1.2: Tratamiento de errores.	I	V	V	V	V	En caso de que el usuario deje el campo Soporte vacío y presiones el botón guardar el sistema envía el siguiente mensaje “El formulario no	No se guarda el material audiovisual en el servidor de medias.

CAPÍTULO 4: “Construcción de la solución propuesta”.

						está listo para el envío.”	No se guarda la información referente a él en la base de datos.
EC 1.2: Tratamiento de errores	V	V	I	V	V	En caso de que el usuario deje el campo de la restricción vacío y seleccione la opción cargar el sistema envía el siguiente mensaje “El formulario no está listo para el envío.” El campo es obligatorio.	No se guarda el material audiovisual en el servidor de medias. No se guarda la información referente a él en la base de datos.
EC 1.2: Tratamiento de errores	V	V	V	I	V	En caso de que el usuario deje el campo de la ruta vacío y seleccione la opción cargar el sistema envía el siguiente mensaje “El formulario no está listo para el envío.” El campo es obligatorio.	No se guarda el material audiovisual en el servidor de medias. No se guarda la información referente a él en la base de datos.
EC 1.2: Tratamiento de errores	V	V	V	V	I	En caso de que el usuario deje el campo del descripción vacío y seleccione la opción cargar el sistema envía el siguiente mensaje “El formulario no	No se guarda el material audiovisual en el servidor de medias. No se guarda

CAPÍTULO 4: “Construcción de la solución propuesta”.

						está listo para el envío.” El campo es obligatorio.	la información referente a él en la base de datos.
EC 1.2: Tratamiento de errores	V	V	V	V	V	En caso de que el usuario deje el campo del proyecto vacío y seleccione la opción cargar el sistema envía el siguiente mensaje “El formulario no está listo para el envío.” El campo es obligatorio.	No se guarda el material audiovisual en el servidor de medias. No se guarda la información referente a él en la base de datos.

4.10 Conclusiones

En el presente capítulo se obtuvo como resultado una propuesta de la solución del sistema, solución que servirá de guía para la implementación del Subsistema de Control de Ingesta que se desarrolla. Se desarrollaron los diagramas de clases del diseño así como el de modelo de datos, el diagrama de despliegue y el de componentes, estos fueron realizados basados en el patrón Modelo-Vista-Controlador (MVC). Con el diagrama de despliegue se pretende mejorar el entendimiento de la distribución física del sistema y lógica del sistema. Todos estos artefactos anteriormente planteados servirán para guiar el proceso de desarrollo del software que se obtendrá como resultado de la investigación, además que permiten esclarecer el diseño de la aplicación y permiten que el desarrollo de la misma sea más rápido, organizado y mejor estructurado.

CONCLUSIONES GENERALES

Con el desarrollo del presente trabajo de diploma se arriba a las siguientes conclusiones:

- La aplicación elimina el problema del registro de las fallas en hojas de papel que puedan ser detectadas durante la digitalización de un video en el ICRT.
- El estudio de las soluciones existentes permitió afirmar que no es posible utilizar ninguna por causa de las licencias y términos legales bajo las cual están desarrolladas.
- La metodología de desarrollo, las herramientas y los lenguajes utilizados, brindaron el soporte necesario para lograr una aplicación que cumpla con los requisitos deseados.
- El trabajo ingenieril inherente a la aplicación ha permitido documentar de manera detallada los artefactos generados en cada flujo de trabajo propuesto por la metodología escogida, con el objetivo de reutilizarlo en futuras versiones.
- El desarrollo del sistema mediante el patrón arquitectónico MVC, garantiza la flexibilidad y extensibilidad del mismo, evitando de esa forma la rápida obsolescencia.

Finalmente dados los aspectos antes mencionados se concluye que el objetivo general para el presente trabajo de diploma ha sido cumplido, obteniéndose como resultado una aplicación Web que permite la ingesta de los nuevos materiales audiovisuales al sistema que se desarrolla en el proyecto SCCM, además de mejorar el control y la organización del proceso de gestión de fallas originadas durante la digitalización de materiales audiovisuales en el ICRT.

RECOMENDACIONES

Teniendo en cuenta el resultado obtenido en el presente trabajo de diploma se recomienda:

- Mejorar el diseño de las interfaces de usuario de la aplicación con el fin de lograr una mejor interactividad con el usuario.
- Perfeccionar las funcionalidades que brinda el sistema con el objetivo de lograr una aplicación óptima.
- Continuar añadiéndole nuevas funcionalidades a la aplicación con el fin de obtener mejoras en futuras versiones.
- Realizar un documento de ayuda en formato digital para usuarios inexpertos.

GLOSARIO DE TÉRMINOS

Audiovisual: Se refiere a las imágenes en movimiento y/o a los sonidos grabados, registrados en película, cinta magnética, disco o cualquier otro medio actualmente conocido o por inventar.

BPM: (Business Process Management) o Administración de procesos empresariales.

Digitalizar: Acción de convertir en digital (aportando valores en bits 0 y 1) cualquier tipo de información, ya sea gráfica, de audio, vídeo, vídeo en movimiento, etcétera.

DOM: Document Object Model o Modelo de Objetos del Documento.

Gestión: implica al conjunto de trámites que se llevan a cabo para resolver un asunto o concretar un proyecto, se refiere al conjunto de actividades que desarrollan, movilizan y motivan al personal empleado que una empresa necesita para su éxito.

Ingesta: Es el proceso de entrada, al Sistema de Captura y Catalogación de Medias, de los materiales audiovisuales en un formato final, o sea ya digitalizados, que serán usados para su posterior organización, edición, catalogación, indexación, almacenamiento, etc.

Materiales Audiovisuales: procura abarcar el máximo de formas y formatos. Las imágenes en movimiento constituyen la clásica forma de material audiovisual, puede incluir sonido o no.

OMG: Object Management Group u Objeto de Administración de Grupos.

Proceso: De acuerdo al diccionario de la Real Academia Española (RAE), el concepto hace referencia a la acción de ir hacia adelante, al transcurso del tiempo, al conjunto de las fases sucesivas de un fenómeno natural o de una operación artificial.

SGML: Standard Generalized Markup Language o Estándar de Lenguaje de Mercado Generalizado.

TIC: Tecnología de la Información y las Comunicaciones.

Televisión: Es un sistema de transmisión de imágenes y sonido a distancias a través de ondas.

BIBLIOGRAFÍA CITADA

1. Saludalia. *Saludalia*. [En línea] [Citado el: 10 de 11 de 2010.] <http://www.saludalia.com/servlet/ServletConsultaDefinicion?idTermino=934&Termino=Ingesta>.
2. que-significa. *que-significa*. [En línea] [Citado el: 10 de 11 de 2010.] <http://que-significa.com.ar/significado.php?termino=ingesta>.
3. Educacion. *Educacion*. [En línea] [Citado el: 10 de 11 de 2010.] <http://ares.cnice.mec.es/informes/13/contenido/12.htm>.
4. Suárez, Yanly García. *Diseño de la base de datos para el Grupo de Calidad de la facultad 9*. Ciudad de La Habana : s.n., 2010.
5. Unesco. *Unesco*. [En línea] [Citado el: 20 de 11 de 2010.] <http://www.unesco.org/webworld/publications/philos/philos3.htm>.
6. Mastermagazine. *Mastermagazine*. [En línea] 2004. [Citado el: 20 de 11 de 2010.] <http://www.mastermagazine.info/termino/4624.php>.
7. Wordreference. *Wordreference*. [En línea] 2010. [Citado el: 20 de 11 de 2010.] <http://www.wordreference.com/definicion/digitalizar>.
8. Tedral. *Tedral*. [En línea] 29 de 03 de 2010. [Citado el: 20 de 11 de 2010.] http://www.tedral.com/index.php?option=com_content&view=article&id=1280:nab-2010&catid=106:newses&Itemid=242.
9. Panorama Audiovisual. *Panorama Audiovisual*. [En línea] 19 de 02 de 2010. [Citado el: 20 de 11 de 2010.] <http://www.panoramaaudiovisual.com/2010/02/19/rtve-digitalizara-un-millon-de-cintas-con-ayuda-de-tedral/>.
10. Tedral. *Tedral*. [En línea] 29 de 3 de 2010. [Citado el: 20 de 11 de 2010.] http://www.tedral.com/index.php?option=com_content&view=article&id=101&Itemid=65.
11. Wordpress. *Wordpress*. [En línea] 24 de 3 de 2010. [Citado el: 21 de 11 de 2010.] <http://isid.wordpress.com/tag/videoma/>.
12. ISID. *ISID*. [En línea] 2006. [Citado el: 21 de 11 de 2010.] http://www.isid.es/spanish/product/producto_v02.htm.
13. Santos., Carelys Suárez Arencibia y Anibal Santos. *SIGNUC: Sistema para la gestión de la información de muestras de núcleos en el CEINPET*. Ciudad de La Habana : s.n., 2009.
14. Cientec. *Cientec*. [En línea] [Citado el: 17 de 01 de 2011.] <http://www.cientec.com/analisis/ana-uml.html>.
15. Enrique Hernández Orallo. *Disca*. *Disca*. [En línea] [Citado el: 18 de 01 de 2011.] <http://www.disca.upv.es/enheror/pdf/ActaUML.PDF>.
16. Apple. *Apple*. [En línea] 2011. [Citado el: 24 de 01 de 2011.] http://translate.google.com/translate?hl=es&langpair=en|es&u=http://www.apple.com/downloads/macosx/development_tools/visualparadigmforumlenterpriseedition.html.
17. María Sierra. *Trabajando con Visual Paradigm for UML*. *Trabajando con Visual Paradigm for UML*. [En línea] [Citado el: 24 de 01 de 2011.] <http://personales.unican.es/ruizfr/is1/doc/lab/01/is1-p01-trans.pdf>.

18. Definicion.org. *Definicion.org*. [En línea] [Citado el: 24 de 01 de 2011.] <http://www.definicion.org/lenguaje-de-programacion>.
19. Valdés, Damián Pérez. *Maestros del Web. Maestros del Web*. [En línea] [Citado el: 26 de 01 de 2011.] <http://www.maestrosdelweb.com/principiantes/los-diferentes-lenguajes-de-programacion-para-la-web/>.
20. I.E.S Gaviota. *I.E.S Gaviota*. [En línea] [Citado el: 25 de 01 de 2011.] <http://www.juntadeandalucia.es/averroes/iesgaviota/informatica/html.html>.
21. Opensuse. *Opensuse*. [En línea] 2011. [Citado el: 07 de 02 de 2011.] <http://es.opensuse.org/Postgresql#Licencia>.
22. Javier Eguiluz. *symfony.es. symfony.es*. [En línea] 2011. [Citado el: 07 de 02 de 2011.] <http://www.symfony.es/que-es-symfony/>.
23. Anieto2k. *Anieto2k*. [En línea] 31 de 03 de 2009. [Citado el: 07 de 02 de 2011.] <http://www.anieto2k.com/2009/03/31/dojo-es-el-framework-javascript-mas-rapido/>.
24. Tordesillas, Rodrigo Testillano. *Estudio de frameworks para la creación de interfaces gráficas. Estudio de frameworks para la creación de interfaces gráficas*. [En línea] [Citado el: 08 de 02 de 2011.] http://www.google.com.cu/url?sa=t&source=web&cd=7&ved=0CD4QFjAG&url=https%3A%2F%2Fforja.rediris.es%2Fdocman%2Fview.php%2F859%2F1322%2FFrameworks_Comparativa_forja.pdf&rct=j&q=caracteristicas%20de%20dojo%20como%20framework&ei=oQZiTdSEJlfd4gbPxbXVCQ&usq=AFQ.
25. Servicio de Difucion de la Creacion Intelectual. *Servicio de Difucion de la Creacion Intelectual*. [En línea] Universidad Nacional de La Plata, 2011. [Citado el: 25 de 02 de 2011.] <http://sedici.unlp.edu.ar/ARG-UNLP-TPG-0000000047/868.pdf>.
26. CLAUDIA LORENA ESCOBAR. Grupo UML. *PATRONES DE MODELADO*. [En línea] 04 de junio de 2006. [Citado el: 23 de 05 de 2011.] <http://umlpnamericana.blogspot.com/2006/06/patrones-de-modelado.html>.
27. Penadés, Patricio Letelier y M^a Carmen. *willydev.net. willydev.net*. [En línea] 23 de 11 de 2010. [Citado el: 15 de 1 de 2011.] <http://willydev.net/de-las-m%C3%A9todolog%C3%ADas-%C3%A1giles-en-el-desarrollo-de-software-2010-11-23.aspx>.
28. Sanchez, María A. Mendoza. *Informatizate. Metodologías de desarrollo*. Perú S.A.C. : Microsoft Certified Professional, 2004.
29. EcuRed. *EcuRed*. [En línea] 07 de 02 de 2011. [Citado el: 21 de 01 de 2011.] http://www.ecured.cu/index.php/Rational_Rose_Enterprise_Edition.
30. Anadic Sinaloa. *Anadic Sinaloa*. [En línea] 18 de 08 de 2009. [Citado el: 2 de 02 de 2011.] http://www.anadicsinaloa.com/index.php?option=com_content&view=article&id=207:caracteristicas-mysql&catid=16:anadic-sinaloa&Itemid=33.
31. David Masip. *Desarrolloweb.com. Desarrolloweb.com*. [En línea] 2002. [Citado el: 02 de 02 de 2011.] <http://www.desarrolloweb.com/articulos/840.php>.