

Universidad de las Ciencias Informáticas

Facultad 8



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

TÍTULO: SISTEMA DE GESTIÓN DE SOLICITUDES DE RECURSOS AUDIOVISUALES

Autores:

**Lisbell Manresa Núñez
Esneiker Enriquez Cabrera**

Tutor:

Ing. Yaillet Martínez Pérez

Ciudad de La Habana, Julio de 2007.

“Año 49 de la Revolución”

DECLARACIÓN DE AUTORÍA

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

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

Firma del Autor

Firma del Autor

Firma del Tutor

“Sólo el conocimiento que llega desde dentro es el verdadero conocimiento”

Sócrates

Agradecimientos

De Esneiker

A mis padres, como recompensa por el amor, la comprensión y el cariño, por todos los años de dedicación y de esfuerzo. Un millón de gracias por confiar en mí.

A Osnier (mi hermano), por creer siempre en mí, por la preocupación, el interés, el amor, el orgullo, la comprensión, por ser el mejor hermano del mundo. Para seguir siendo tu ejemplo es que me esfuerzo cada día un poco más.

A Michel (mi otro hermano), para que te inspires a ser ingeniero.

A Eloy: por todos los consejos que me haz dado y por tenerme presente.

A toda mi familia, por el apoyo que me han dado, mis primos, mis tías (Magali y Plácida)..., a todos ellos por interesarse en mi.

A Yudeisis (mi novia), por tu amor, tu dedicación, tu apoyo, tu sacrificio, para ti el mayor beso del mundo, te quiero con la vida.

A Reynaldo Roselló, por tu ayuda, tu preocupación, tu apoyo, por todas las ideas que me diste. Por ser mi amigo. Por todo un millón de gracias.

A Yaillet (nuestra tutora), por habernos confiado el desarrollo de este trabajo.

A todas las personas, que de una forma u otra han dado su apoyo para ayudarme a desarrollar el trabajo.

De Lisbell

Mima: A mi viejita linda, esa personita que supo quererme desde que nací y me cuidó como si fuera otra bebé suya, aunque no pude cumplir tu sueño de ser doctora, pues no estaba en mis planes, no quisiera que te sintieras triste pues doctora o informática siempre seré tu nieta querida. Gracias mima por pensar en mí, quererme mucho, porque sé que desde la casa rezas por mí para que salga bien y pueda lograr todos mis sueños, por llorar y estar junto a mí cuando me han tratado mal o roto el corazón.

Tata: Gracias por tratar de complacerme en todos mis deseos, por tratar de darme todo lo que necesito, por ser mi madre y padre al mismo tiempo. De correr con todos mis problemas y buscarme la mejor

solución. Gracias Tata por darme la vida, estar en mi vida, por dejar todo lo que es importante para ti solo para estar al lado mío.

Tía: Yo no diría que eres esa persona sino más, eres como mi abuela o mi mamá. Gracias por estar atenta a cada llamada mía, escuchando un llanto, una queja o simplemente una riza. Por defenderme cuando no he tenido nadie que lo haga y ponerme en muchas ocasiones por encima de tus hijos.

Sonia: A pesar de conocerte hace tan poco tiempo, te haz portando como si fueras mi mamá. Te agradezco cada sonrisa, abrazo o regaño ante cualquier acción mía. Gracias por considerarme una hija, por acogerme en tu familia, no sé cuantas cosas te pudiera decir pero sé que con mis palabras no serían suficientes para agradecerte, solo te puedo decir que al lado tuyo, Rubén y Rigo me he sentido como una más, no como una nuera sino como una hija porque ese siempre ha sido mi sueño vivir en familia con mis padres y cada vez que veo o comparto en familia siempre viene a mi cabeza esos pensamientos. Gracias por brindarme esos momentos de felicidad.

Yaillet: Gracias por tu voto de confianza en mí, por haberme tenido presente cuando necesitaste una persona capaz de desempeñar la tarea que necesitabas. A ti que haz aguantado todos mis reproches y chiquilladas, por aconsejarme y guiarme en el mejor camino de la vida.

A mi papá: A ti padre aunque no me has enseñado a como vivir, sé que en el fondo me quieres y aunque a veces no estás junto a mí no te reprocho nada, porque eres mi papá. Te quiero mucho y lloro cuando tú no estás para aconsejarme. Pese a todo también formas parte de mí por eso te agradezco y espero que estés contento de tener una hija ingeniera.

Son más de mil momentos que hemos compartido, mil instantes juntos y muchas palabras las que nos hemos dicho. Gracias **Rigo** por ser más que mi amigo, gracias por los consejos, regaños como si fueras mi padre, críticas y alegrías, yo sé que todo ha sido para bien. Quizás no nos volvamos a ver pero procuraré meterme en tus sueños y en tus pensamientos, saber tus esperanzas y estar ahí cada instante, para ayudarte y compensar tus actos hacia mí.

Gracias a todos mis amigos (Yaikiel, Andy, Yudel, Yaiser, Duniel) que de una forma u otra me han ayudado y brindado su apoyo durante estos hermosos 5 años universitarios.

También queremos agradecer a nuestros compañeros de aula por el apoyo. A La Universidad de las Ciencias Informáticas, a La Revolución Cubana y en especial a nuestro comandante Fidel Castro Ruz.

A nuestros Padres...

Resumen

Este trabajo se propone estudiar la propuesta de un sistema informático que permita informatizar los procesos del flujo de trabajo de Gestión de Recursos Audiovisuales de La Dirección de Comunicación Audiovisual (DCAV) de La Universidad de las Ciencias Informáticas. Estos procesos se realizan de forma manual y no responden totalmente a las necesidades de los clientes. El desarrollo de un sistema informático ayudaría a aumentar el control de los procesos actuales. Por tanto el objetivo concreto de este trabajo radica en desarrollar una aplicación informática que proporcione un mayor control sobre los procesos que se llevan a cabo en el flujo de trabajo de la DCAV y que facilite la realización de las tareas por parte del personal que labora con las solicitudes de medias. En el documento se presentan los resultados obtenidos a partir de la investigación realizada. Se identifican y describen los procesos que se llevan a cabo en la DCAV, haciendo énfasis en los procesos que se desean informatizar. En el documento se demuestra la necesidad de implementar un sistema que resuelva las necesidades que existen actualmente. Se realiza un análisis comparativo de las metodologías y tecnologías existentes y se seleccionan las más apropiadas para elaborar la propuesta de solución. Los resultados obtenidos en el trabajo son el análisis y diseño y la implementación del sistema.

Índice

Introducción	1
Capítulo 1. Fundamentación.....	4
1.1 <i>Introducción.</i>	4
1.2 <i>Gestión de Recursos Audiovisuales.</i>	4
1.3. <i>Flujo de Trabajo.</i>	7
1.4 <i>Estudio de las tendencias y tecnologías.</i>	9
1.4.1 Metodologías de Desarrollo de Software.....	9
1.4.2 Lenguajes de programación web.....	13
1.4.3 Servidores web.	17
1.4.4 Sistemas Gestores de Base de Datos.....	17
1.4.5 Estilo de Arquitectura.....	20
1.4.6 Herramientas utilizadas en la implementación de la aplicación.	21
1.5 <i>Conclusiones.</i>	24
Capítulo 2. Características del Sistema.....	25
2.1 <i>Introducción.</i>	25
2.2 <i>Problema y situación problemática.</i>	25
2.3 <i>Objeto de automatización.</i>	26
2.4 <i>Información que se maneja.</i>	26
2.5 <i>Propuesta del sistema.</i>	27
2.6 <i>Modelo de Negocio.</i>	28
2.6.1 Actores del Negocio.....	28
2.6.2 Trabajadores del Negocio.....	28
2.6.3 Diagrama de Casos de Uso del Negocio.....	29
2.6.4 Descripción del proceso de negocio.....	30
2.6.5 Diagrama de Clases del Modelo de Objeto.	31
2.6.6 Diagrama de Actividades.....	32
2.6.7 Reglas del Negocio.....	32
2.7 <i>Requisitos funcionales del sistema.</i>	33
2.8 <i>Requerimientos no Funcionales.</i>	35
2.9 <i>Definición de los Casos de Uso.</i>	37
2.9.1 Actores del Sistema.	37

2.9.2 Diagrama de Casos de Uso del Sistema.....	38
2.9.3 Descripción textual de los Casos de Uso del Sistema.....	42
2.10 Conclusiones.....	57
Capítulo 3. Análisis y Diseño del Sistema	58
3.1 Introducción.....	58
3.2 Modelo de Análisis.....	58
3.3 Modelo de Diseño.....	59
3.3.1 Clases Persistentes.....	59
3.3.2 Modelo de Datos.....	59
3.3.3 Diagramas de Secuencia.....	71
3.3.4 Diagramas de Clases Web.....	71
3.4 Patrones de Diseño (Grasp).....	71
3.5 Estándares de Interfaz de Aplicación.....	72
3.6 Estándares de Codificación.....	72
3.7 Formatos de Reportes.....	74
3.8 Tratamiento de errores.....	74
3.9 Seguridad.....	74
3.10 Concepción General de la ayuda.....	75
3.11 Conclusiones.....	75
Capítulo 4. Implementación y Prueba.....	76
4.1 Introducción.....	76
4.2 Modelo de Despliegue.....	76
4.3 Diagramas de Componentes.....	77
4.4 Modelo de prueba.....	78
4.5 Conclusiones.....	92
Conclusiones Generales.....	93
Recomendaciones.....	95
Referencias Bibliográficas	96
Bibliografía.....	98
Glosario de Términos.....	99

Introducción

En La Universidad de las Ciencias Informáticas (UCI) se desarrollan importantes proyectos productivos que ayudan al desarrollo de la universidad, además se realizan otros que están relacionados con diferentes instituciones de nuestro país ya sean vinculados a la medicina, la educación o el deporte.

En la actualidad el desarrollo de las Tecnologías de la Informática y las Comunicaciones (TIC) ha permitido que se incrementen las demandas de productos de software educativo a nivel nacional e internacional. Nuestra universidad no está exenta, pues presenta una fuerte línea de desarrollo en este sentido. Existe el departamento de Software Educativo, el cual es el encargado de controlar el proceso de desarrollo de los productos de multimedia que le asignen a la UCI.

El trabajo realizado en la DCAV es muy incómodo ya que se maneja un gran volumen de información y no existe un sistema informático que mejore la realización de las actividades. Además no se cuenta con un mecanismo eficiente que notifique a los clientes cuando los recursos audiovisuales de sus solicitudes han sido gestionados completamente y por tanto se produzca un atraso en los proyectos que requieran de los recursos audiovisuales producidos.

Otro inconveniente que existe en la DCAV es que como no se cuenta con un sistema de aviso, no se tiene conocimiento sobre en qué estado se encuentra cada uno de los recursos audiovisuales.

Una desventaja importante en el proceso de Gestión de Recursos Audiovisuales es que existe un número reducido de personas que son las encargadas de realizar todas las tareas de la DCAV correspondientes a la gestión de los recursos audiovisuales. Esto provoca que se atrasen las solicitudes y que en algunos casos no se cuente con un producto de alta calidad.

Por todo lo planteado se decidió informatizar el proceso de Gestión de Recursos Audiovisuales para alcanzar un mayor control y eficiencia en el mismo. Control porque todas las operaciones realizadas a una solicitud quedarían registradas en un mecanismo persistente, de forma que se pueda conocer en todo momento quién realizó determinada actividad sobre algún recurso audiovisual, y eficiente porque un sistema facilitaría todo el trabajo con la documentación de las solicitudes además de mantener una mayor organización en el mismo.

El presente trabajo de diploma surge debido a la necesidad de dar una rápida solución a la situación antes expuesta, por lo tanto el **problema científico** consiste en la no existencia de un sistema capaz de dar

solución a los problemas actuales en el desarrollo de este flujo de trabajo, que permita que el proceso de Gestión de Recursos Audiovisuales se realice de forma eficiente.

Por tanto el **objeto de estudio** de esta investigación es el proceso de desarrollo de Sistemas de Software Educativo Multimedia. Y como **campo de acción** el proceso de Gestión de Recursos Audiovisuales para el desarrollo de Multimedias en La Universidad de las Ciencias Informáticas.

Como **Idea a defender** se plantea que si se implementa un sistema automatizado que permita realizar la gestión de las solicitudes de los recursos audiovisuales con eficiencia y control se podría mejorar la eficacia del proceso de Gestión de Recursos Audiovisuales y la calidad de los proyectos de Software Educativos Multimedia en la UCI.

El **objetivo general** es el desarrollo de un sistema para La Gestión de las Solicitudes de Recursos Audiovisuales (video, imágenes, sonido, locuciones, animaciones, digitalizaciones) en la UCI, haciendo uso de técnicas y herramientas para la implementación de una aplicación informática que permitirá realizar la gestión de los recursos audiovisuales de forma eficiente y controlada.

En correspondencia con el objetivo general definido, se plantean los siguientes **objetivos específicos**:

- Estudio del arte sobre el tema a tratar.
- Realizar el Análisis y Diseño del sistema.
- Implementar un sistema automatizado para el desarrollo del flujo de trabajo de gestión de recursos audiovisuales en La Universidad de las Ciencias Informáticas.

Para dar cumplimiento a los objetivos específicos definidos en el trabajo se deben realizar las siguientes **tareas**:

- Estudiar el proceso de Gestión de Recursos Audiovisuales en la UCI.
- Analizar las tendencias y tecnologías actuales para el desarrollo de aplicaciones informáticas.
- Seleccionar la metodología más adecuada para el análisis y diseño del sistema.
- Seleccionar las herramientas apropiadas para diseñar la aplicación.
- Realizar el Análisis y Diseño del sistema.
- Diseñar una base de datos que soporte las funcionalidades del sistema informático que se pretende desarrollar.
- Realizar la implementación del sistema.

Con el desarrollo del presente trabajo de diploma se pretende obtener un sistema que mejore la eficiencia y el control de la gestión de recursos audiovisuales en la universidad.

El documento está compuesto por 4 capítulos.

El **capítulo 1** “Fundamentación Teórica” está dedicado al estudio del arte del tema relacionado con la investigación. Se hace referencia a las tendencias y tecnologías actuales que se utilizan en la realización de aplicaciones informáticas, así como las distintas metodologías y lenguajes de programación que existen para el diseño e implementación.

El **capítulo 2** “Características del Sistema” especifica la problemática, el objeto de estudio y el objeto de automatización. En este capítulo se describen los procesos que se realizan actualmente, a través de un Modelo de Negocio. Se establecen las reglas del negocio. Además se determinan los actores, trabajadores, entidades y casos de uso del negocio. Se definen los requisitos funcionales y no funcionales del sistema. Se realiza el modelo de casos de uso del sistema y se describe cada caso de uso.

El **capítulo 3** “Análisis y Diseño” se realiza el análisis y diseño del sistema. Se construyen los diagramas de clase y secuencia del análisis y del diseño. Se realiza el diagrama de clases persistentes y el modelo de la base de datos. Se especifican los principios del diseño detallando los estándares de codificación, tratamiento de errores, seguridad y ayuda que poseerá el sistema.

El **capítulo 4** “Implementación y Prueba” trata los principios de implementación y prueba del sistema. Se construyen los diagramas de componentes y despliegue. Se describen los casos de prueba del sistema y se realizan las pruebas de caja negra sobre las interfaces.

1

CAPÍTULO Fundamentación Teórica

1.1 Introducción.

En el presente capítulo se realiza un estudio del arte entorno al objeto de investigación y campo de acción. Se hace referencia a las tendencias, técnicas, tecnologías, metodologías y software que se utilizan en la actualidad a nivel nacional e internacional y pudieran formar parte de la propuesta de solución del problema a resolver.

Se hace un estudio de las metodologías de desarrollo de software que permitirán analizar el flujo de trabajo y desarrollar un software como producto final, así como de los servidores, lenguajes de programación web y los gestores de base de datos, de todos ellos se analizan sus características, funciones y ventajas.

Se realiza el análisis de las herramientas auxiliares que posibiliten completar el desarrollo del producto requerido. Finalmente se hace una selección de las herramientas, técnicas y metodologías que permitan solucionar el problema y cumplir los objetivos propuestos en el trabajo.

También se hace un estudio de los flujos de trabajo debido a que se pretende informatizar el flujo de trabajo de La Dirección de Comunicaciones Audiovisual de La Universidad de las Ciencias Informáticas. Esto es necesario para comprender cómo se aplican los flujos de trabajo y qué ventajas ofrecen para las organizaciones.

1.2 Gestión de Recursos Audiovisuales.

Según el trabajo realizado por un colectivo de autores de La Universidad de las Ciencias Informáticas donde proponen una metodología para el desarrollo de Software Educativos Multimedia en dicha universidad, se derivan diferentes procesos, los cuales a su vez definen otros flujos de trabajo que pueden o suelen estar interrelacionados. Este flujo de trabajo se definió siguiendo los principios definidos en la metodología RUP.

Los procesos definidos para el desarrollo del flujo general de trabajo se muestran a continuación:

- Definición del Proyecto.
- Gestión de Requisitos.
- Evaluación Técnica.
- Gestión de Diseño Gráfico.
- Gestión de Recursos Audiovisuales.
- Construcción.
- Gestión de Configuración. [3]

En cada de una de estas etapas se realiza una parte importante del desarrollo del producto. Primeramente se debe definir el proyecto. Luego se realiza un análisis de lo que desea el cliente y se evacuan todas las dudas posibles para obtener una comprensión total del producto a realizar. En la Evaluación Técnica se continúa analizando el proyecto, además de las especificaciones de los productos a obtener, se hace un diseño gráfico en función de los requisitos obtenidos del guión técnico. También se encuentra La Gestión de Recursos Audiovisuales, proceso importante porque es donde se realizan las búsquedas y producciones de recursos audiovisuales de acuerdo a los requisitos del cliente.

Por otro lado el proceso de Construcción es donde se realiza la implementación, revisión técnica y corrección de errores del producto para después realizar la entrega final al cliente.

La definición de procesos, procedimientos y políticas en una empresa son factores de suma importancia para garantizar eficiencia y calidad. Lograr que los procesos productivos se realicen según un marco de trabajo disminuye los costos, el tiempo, brinda información a la empresa para realizar mejoras a partir de los errores encontrados y ayuda grandemente a la mitigación de los riesgos. [3]

La Gestión de Recursos Audiovisuales es un proceso de alta complejidad que se realiza en el proceso de desarrollo de software educativo. Sin embargo, los principales trabajadores de este proceso lo constituyen personal de la DCAV. Esta dirección es la encargada de realizar las operaciones del flujo de trabajo para producir los recursos audiovisuales solicitadas por los diferentes proyectos. A continuación se describe el flujo de trabajo que se realiza durante este proceso:

Solicitud de media

Se realizan las solicitudes de recursos audiovisuales a la DCAV. Estas son realizadas por los Vicedecanos y Líderes de proyectos, obteniéndose como resultado el Documento de Solicitud de Media.

Definición de prioridad

En dependencia del proyecto se le asignará una prioridad a cada solicitud para su gestión que debe guiar a los especialistas de la DCAV. Los participantes son los Especialistas de La Dirección de Software Educativo (DSWE). El resultado es la definición de la prioridad a la solicitud.

Validación de la solicitud

Se validan las planillas de las solicitudes para garantizar que no contengan errores (Se aceptan o se rechazan). En caso de ser aceptada se define un responsable para la gestión de los recursos audiovisuales. El responsable de esta actividad es la DCAV y se obtiene como resultado planillas de solicitudes carentes de errores.

Gestión de Recursos Audiovisuales

Consiste en la búsqueda o producción de los recursos audiovisuales solicitados. Este proceso dentro del flujo de trabajo de la DCAV no es objetivo a informatizar en esta investigación, por tal motivo no se hace un análisis profundo sobre las acciones llevadas a cabo en el mismo.

Entrega de medias

Se copian los recursos audiovisuales en el servidor y se modifica el estado de la planilla. El responsable de ejecutar esta actividad es el Especialista de la DCAV, obteniéndose la copias de los recursos audiovisuales en el servidor de medias y la planilla de medias entregadas.

Validación de entregas

Se validan las planillas entregadas (Se aceptan o se rechazan los recursos audiovisuales). Si no se aceptan se especifican las razones por la que rechazan.

Etapa 1: Validación Pedagógica.

Se realizan las validaciones de contenido, con el objetivo de validar que el contenido de los recursos audiovisuales se ajuste al solicitado. El responsable es el Especialista de DSWE y se obtiene como resultando un listado de recursos audiovisuales no aceptados.

Etapa 2: Validación Técnica.

Se realizan las validaciones técnicas donde se verifica que los recursos audiovisuales se ajusten a los parámetros definidos por los Vicedecanos o Jefes de Líneas y concuerden los nombres con los de las

planillas de la solicitud. Esta actividad es desarrollada por el Líder de proyecto, y se obtiene como resultando un listado de los recursos audiovisuales no aceptados.

Aceptación de los recursos audiovisuales.

Una vez que se validan todos los recursos audiovisuales entregados, se cierra el pedido emitiendo un documento de aceptación de recursos audiovisuales. El responsable de realizar esta acción es el Especialista de DSWE. [3]

Luego de realizar una investigación sobre la búsqueda de algunos sistemas que puedan brindar una solución favorable al proceso de Gestión de Recursos Audiovisuales en la universidad se concluyó que no existe un sistema informático que pueda resolver la necesidad, puesto que dicho proceso es único para la gestión de recursos audiovisuales, y cada una de las actividades desarrolladas son típicas de La Dirección de Comunicación Audiovisual.

1.3. Flujo de Trabajo.

El auge de la informática ha hecho posible que la información que se maneja hoy en día se haya incrementado, los procesos de negocios se hacen muy complejos para su realización de forma manual, esto provoca que se comentan más errores que conducen a la baja calidad de los productos empresariales y la tenencia de problemas con los clientes por falta de información, etc. Para erradicar estos problemas se implementan gestores de bases de datos más potentes para almacenar y consultar grandes volúmenes de información, se trata de mejorar los procesos de negocios, de particionar el trabajo, colaborar mediante el trabajo en equipo, realizar las tareas con un mismo objetivo o meta: la de lograr un producto con calidad, eficiencia, en tiempo y con la mayor aceptación para los clientes.

Para organizar el trabajo en las empresas se adopta el uso de los flujos de trabajo puesto que permiten tener un mayor control sobre los procesos que se desarrollan.

El flujo de trabajo se define como un conjunto de métodos y tecnologías que ofrece facilidades para modelar y gestionar diversos procesos que ocurren dentro de una empresa. Se relaciona con la automatización de los procedimientos donde los documentos, la información o las tareas son pasadas entre los participantes del sistema de acuerdo a un conjunto de reglas previamente establecidas. [1]

Marshak, en el año 1994 planteó lo siguiente: “Es la automatización de los procesos que usamos todos los días para llevar a cabo nuestro negocio. Una aplicación de flujo de trabajo hace automática la secuencia

de acciones, actividades, o tareas usadas para ejecutar el proceso, incluyendo el seguimiento del estado de cada instancia del proceso, así como también las herramientas para manejar el proceso mismo.” (GARCÍA 1999)

El flujo de trabajo surgió por la necesidad de automatizar actividades o procesos que se realicen de forma manual, con el propósito de realizar una mejor gestión de la información, incrementar las búsquedas rápidas y eficientes, perfeccionar la productividad, reducir tiempo en los procesos, obtener un control sobre estos, además de disminuir costos para mejorar los productos y obtener mayor competitividad y aceptación en el mercado.

Proporciona grandes beneficios, no solo se enmarcan en el ahorro de tiempo, sino en otros aspectos importantes tales como:

- Eficiencia en los procesos y estandarización de los mismos.
- Asignación de tareas a las personas.
- Recursos disponibles.
- Diseño de procesos.[1]

El flujo de trabajo facilita la modelación de los procesos de la empresa, permitiendo hacer un análisis y diseño más profundo. No es solo una tecnología que provee el cambio, sino también, proporciona un marco de referencia para el análisis y diseño previo a la implementación de un sistema que involucra la interacción de diversos procesos.

Los procesos dentro de una empresa son el punto de partida de la tecnología flujo de trabajo ya que su objetivo fundamental es automatizarlos. Esto se garantiza con una correspondencia y acoplamiento entre las personas que participan en ello. Para hacer uso de las técnicas de flujo de trabajo se requiere al menos dos personas que trabajen en colectivo y con un objetivo común.

Hoy en día se pueden considerar tres tipos de flujo de trabajo como se refiere a continuación:

- De Producción.
- De Colaboración.
- De Administración.

El Flujo de Trabajo de Colaboración estructura o semi-estructura procesos de negocios donde participan personas para lograr una meta común. Típicamente involucran documentos contenedores de la

información, se sigue la ruta de estos paso a paso, además de las acciones que se toman sobre ellos. Los documentos son la clave, por lo que es esencial para la solución de flujo de trabajo mantener la integridad de los mismos.

El Flujo de Trabajo Administrativo involucra procesos de administración en una empresa tales como órdenes de compra, reportes de ventas, etc. [1]

Las transacciones en una base de datos se consideran la clave en el Flujo de Trabajo de Producción. Se automatizan procesos de negocio que tienden a ser repetitivos (sigue reglas y patrones similares para toda instancia del proceso), bien estructurados y con gran manejo de datos.

Las aplicaciones que utilizan este tipo de flujo de trabajo deben cumplir con los siguientes atributos: [2]

- La velocidad de transferencia es importante en el flujo de trabajo de producción, ya que es la tarea principal de los participantes.
- La flexibilidad de poder cambiar el proceso no suele ser importante. Una vez establecido el flujo, este permanece sin cambio por largo tiempo.
- El Flujo de Trabajo de Producción suele estar circunscrito a un solo departamento, la escalabilidad, o capacidad de crecer no es importante.

El Flujo de Trabajo de Producción se aplica al proceso de Gestión de Recursos Audiovisuales puesto que se trata de automatizar en todo momento procesos que tienden a ser repetitivos, además se manipulan grandes volúmenes de información. También permite que los trabajadores implicados puedan realizar las tareas con mayor agilidad y en menor tiempo proporcionándoles desarrollar otras actividades.

1.4 Estudio de las tendencias y tecnologías.

1.4.1 Metodologías de Desarrollo de Software.

Hoy en día se habla con mayor énfasis acerca de cómo mejorar la calidad de los proyectos de desarrollo de software, realizarlos en poco tiempo y carentes de errores, tratando siempre de proporcionar respuestas correctas a las solicitudes de los clientes. Para ello se crean las metodologías de desarrollo de software, las cuales se clasifican en dos grandes grupos, las que basan su funcionamiento en los llamados métodos pesados que tratan de enfatizar su trabajo en el orden y la documentación y los

métodos ligeros (también se les llaman ágiles) que son los que trabajan en una interacción directa y estrecha entre las personas que intervienen en el proceso de desarrollo de software.

En el desarrollo de un software la metodología define:

- ¿Quién debe hacer cada actividad?
- ¿Qué se debe hacer?
- ¿Cuándo hacerla?
- ¿Cómo se debe hacer el software?

Existen diferentes metodologías para el desarrollo de un proyecto de software, dentro de las más destacadas se encuentran:

- Rational Unified Process (RUP)
- XP
- FDD

No existe una metodología universal para el desarrollo de un software, sino que de acuerdo con las características de cada proyecto, equipo de trabajo y recursos se exigirá un tipo de metodología específica que brinde la mejor solución al problema que requiere solución.

Programación extrema (XP).

La metodología de desarrollo de software XP aparece hace algún tiempo como una disciplina de La Ingeniería del Software. Constituye una nueva manera de ayudar a los programadores a desarrollar excelentes proyectos de software. Esta metodología se basa fundamentalmente en la simplicidad y la acumulación de códigos.

Dentro de los objetivos principales de XP se pueden citar los siguientes: la satisfacción del cliente, donde trata de dar al cliente el software que necesita y cuando lo necesita. Por este motivo debe responder muy rápido a las necesidades del cliente, incluso cuando los cambios sean al final del ciclo de la programación.

El segundo objetivo es potenciar al máximo el trabajo en grupo, o sea, agrupar todas las fuerzas del equipo de desarrollo del software, los jefes de proyecto, los clientes y desarrolladores que están involucrados en el mismo. Esta metodología define cuatro variables para proyectos de software: coste, tiempo, calidad y ámbito. [4]

La Programación Extrema plantea que en el desarrollo del software lo más importante son las personas que trabajan en él, que se debe conocer lo que realmente el software tiene que hacer para que responda a la solicitud del cliente, determinar en qué tiempo se termina el software y por lo tanto qué se debe realizar primero. Debe existir una plena comunicación con el jefe de proyecto para saber como se organiza el trabajo del equipo y tener un control de las versiones del proyecto de software.

Además esta metodología a pesar de que tiene una gran aceptación en programadores presenta muchas críticas en su forma de trabajo. Un ejemplo de ello es el mito de las 40 horas semanales, pues se plantea que es un lujo para las exigencias del mercado. Otra de las desventajas es que solo puede funcionar con programadores muy buenos, con una basta experiencia en la materia. XP es más una filosofía de trabajo que una metodología. Está diseñado para grupos de pequeños programadores, más de 10 programadores resultaría complicado. XP es una metodología liviana y ágil, está orientada más a las personas que a los procesos. [4]

XP es cercano al desarrollo y se basa en UserStories. El código fuente pertenece a todo el equipo de trabajo, pero una desventaja de esta metodología lo constituye la carencia de documentación. [5]

Desarrollo Guiado por la Funcionalidad (FDD).

La metodología FDD se utiliza en proyectos pequeños de corta duración. Su funcionamiento está basado en iteraciones cortas, produciendo un software que permite monitorizar los procesos en cuanto al avance obtenido. Las iteraciones se deciden a través de las funcionalidades que son pequeñas partes del software.

La metodología FDD se divide en 5 fases:

- Desarrollo de un modelo general.
- Construcción de una lista de funcionalidades.
- Plan de releases en base a las funcionalidades a implementar.
- Diseñar en base a las funcionalidades.
- Implementar en base a las funcionalidades. [5]

FDD define métricas para controlar cómo va el desarrollo del proyecto y que se pueda estimar mejor el proyecto en un futuro.

Por otro lado esta metodología presenta su debilidad en la necesidad de tener en el equipo de trabajo miembros con experiencia que marquen el camino a seguir desde el principio, con la elaboración del modelo global, no es tan ágil como XP. [5]

Al contrario de XP, el código fuente tiene propietario. Los equipos varían en cuanto a la funcionalidad a implementar. El conocimiento de la aplicación se reparte a través del trabajo en equipo y revisiones ya que dicha metodología presenta una jerarquía dentro del equipo de desarrollo. [5]

Proceso Unificado de Desarrollo de Software (RUP).

El proceso de desarrollo de RUP se caracteriza por ser:

Dirigido por casos de uso: Define que los casos de usos se refieren a los deseos de los usuarios, los cuales se obtienen a partir de la modelación del negocio y la captura de requisitos. Los casos de uso guían el desarrollo del software.

Centrado en la arquitectura: Porque en la arquitectura se muestra la visión completa que debe tener el sistema propuesto y se representa en un lenguaje común entre los integrantes del equipo de trabajo y los usuarios. Aquí se describen los elementos del modelo que son primordiales para su desarrollo, las bases del sistema que son importantes para comprenderlo, desarrollarlo y producirlo económicamente.

Iterativo e Incremental: Porque propone que cada fase se desarrolle en iteraciones, que se pueda dividir en pequeños proyectos para mejorar su comprensión y desarrollo. Estas iteraciones están presentes en las actividades de los flujos de trabajo, las cuales hacen referencia a pasos en los distintos flujos de trabajo de la metodología del RUP, y los incrementos al crecimiento del producto.

Las actividades en RUP se agrupan en grupos lógicos, definiéndose 9 flujos de trabajo principales y 4 fases. Los 6 primeros flujos de trabajo se conocen como flujos de trabajo ingenieriles (Modelación del Negocio, Requerimientos, Análisis y Diseño, Implementación, Prueba, e Instalación), mientras que los restantes 3 se denominan de apoyo (Administración de Proyectos, Administración de Configuración y Cambios, y el tercero, Ambiente). Las fases se denominan (Inicio, Elaboración, Construcción y Transición).

RUP posee potentes herramientas que ayudan al desarrollo de sistemas de software, como por ejemplo Rational Rose Enterprise Edition que permite representar los diferentes diagramas que se generan en cada flujo de trabajo, o el Project con el cual se pueden definir las tareas, recursos y responsabilidades a

cada integrante del equipo de desarrollo. Además RUP utiliza el UML (Lenguaje Unificado de Modelado) para visualizar, especificar, construir y documentar los artefactos que se obtienen en el desarrollo del software.

Selección de la metodología a utilizar.

Las metodologías analizadas anteriormente poseen pocas similitudes entre ellas, pues XP y FDD son métodos ágiles orientados fundamentalmente a los clientes e iteraciones pequeñas; mientras que RUP es una metodología pesada. Además muchas personas consideran que XP y FDD están dentro de RUP por la complejidad que presenta el mismo. Por otro lado XP es más orientado a las personas que a los procesos y posee poca documentación, mientras que FDD tiene su deficiencia en que los participantes deben tener cierta experiencia en el trabajo, otra desventaja es que el código fuente no es libre.

Según el análisis realizado sobre cada una de las metodologías planteadas se ha decidido que en el desarrollo de la propuesta de proyecto de software para la gestión de recursos audiovisuales en el desarrollo de software educativos multimedia se utilizará la metodología RUP que basa su trabajo fundamentalmente en la documentación del software, por ejemplo el Plan de desarrollo, de calidad e iteración, con cada uno de ellos se puede llevar un control del desarrollo del software para saber el estado de cada fase y detectar posibles errores que puedan salir surgir durante cada flujo de trabajo y así poder corregirlos. También define una estructura de roles para repartir el trabajo del equipo, cada rol tiene funciones específicas a desempeñar, pero todo tiene un objetivo común, el de lograr con un excelente trabajo en colectivo: la construcción de un software con la mejor calidad y eficiencia que pueda resolver los problemas identificados con el cliente.

1.4.2 Lenguajes de programación web.

Hoy en día el desarrollo que ha tenido Internet ha contribuido al intercambio de información utilizando el protocolo HTTP (Hypertext Transfer Protocol, Protocolo de Transferencia de Hipertexto) que permite transferir la información de los sitios web. Cualquier individuo desde diferentes latitudes puede acceder a infinidad de sitios a través del servidor más conocido La World Wide Web (WWW) por medio del modelo cliente/servidor. Este gran avance tecnológico impone que se desarrollen diferentes y nuevos lenguajes de programación.

Los lenguajes de programación se clasifican en lenguajes del lado del cliente o del lado del servidor. Los lenguajes del lado del servidor son lenguajes registrados, ejecutados e interpretados por el servidor y se

envían al cliente en un formato evidente para él (PHP, ASP, Perl). Los lenguajes del lado del cliente (JavaScript, Visual Basic Script) son interpretados en el navegador del cliente.

ASP.NET (Active Server Pages).

ASP.NET es una versión mejorada del lenguaje de programación ASP, con nuevas funcionalidades y servicios que proporcionan un mejor funcionamiento del mismo. ASP.NET se ha diseñado para funcionar sin problemas con editores HTML WYSIWYG y otras herramientas de programación como Microsoft Visual Studio .NET, además pueden utilizar formularios web o servicios Web XML o combinarlos entre ellos. [6]

ASP.NET es una tecnología que mejora el rendimiento y escalabilidad, ofreciendo acceso a lenguajes compilados. Está formado por una biblioteca de clases, incorporada como parte del .NET Framework por lo que se soportan eventos, controles y funciones de caché. [6]

El lenguaje ASP.NET presenta un código sencillo y se puede apreciar la división real entre interfaz y lógica de trabajo, pero también hay que señalar que este modelo no se rige tan claramente como antes por los principios de la programación orientada a objetos (es menos intuitivo desde este punto de vista), es más difícil de mantener sincronizado el código con la interfaz. [7]

PHP (Hypertext Preprocessor).

PHP es un potente lenguaje de programación del lado del servidor, su función principal es generar páginas de manera dinámica. Este lenguaje de programación cuenta con un elevado número de desarrolladores en todo el mundo debido a su facilidad y robustez. Se ha ido desarrollando desde su surgimiento hasta la actualidad. En cada versión se mejoran sus funcionalidades y servicios por cuanto ha ido incrementándose su aceptación. Actualmente este lenguaje se encuentra en su versión PHP5.

Las características más distinguidas son:

- Libre, cualquier persona puede utilizarlo sin necesidad de disponer de una licencia.
- Abierto, se puede modificar el código fuente y añadir nuevas funcionalidades.
- Código fuente disponible, el código fuente se encuentra disponible en la red y se puede descargar cuando se estime conveniente.
- Diseñado para la web, es un lenguaje de programación muy eficiente para desarrollar aplicaciones web.

- Multiplataforma, porque tiene soporte para la gran mayoría de las plataformas existentes en la actualidad.
- Soporte para varios servidores web, se integra con gran número de servidores web, incluso con servidores propietarios.
- Soporte nativo para prácticamente cualquier base de datos.
- Amplia documentación distribuida en la red.
- Perfecta integración del Apache-PHP-MySQL.
- Sintaxis clara y bien definida.
- Sencillo de aprender y utilizar.
- Seguro, debido a que existen muchos desarrolladores de este lenguaje en todo el mundo constantemente se trabaja en mejorar la seguridad del lenguaje.
- Amplia base de usuarios.
- No depende de un único proveedor de servicios.[8]

PHP es un lenguaje de programación capaz de procesar documentos HTML, esto significa que se pueden realizar operaciones antes de mostrar contenido en HTML, esto lo hace ideal para desarrollar sitios web dinámicos. Contiene funciones para trabajar virtualmente con todas las tecnologías para web existentes en la actualidad.

Motores de Plantillas.

Cuando se diseña una aplicación web se deben utilizar plantillas que faciliten el trabajo del diseñador y del programador. Con SMARTY se pueden crear rápidamente grandes cantidades de documentos parecidos entre sí, simultáneamente, separando el lado lógico (scripts, funciones y toda la programación) del lado visual. [9]

SMARTY es un motor de plantillas para PHP y permite separar la presentación lógica y el contenido de la aplicación. Permite que el diseñador del sitio web pueda realizar cambios pertinentes al diseño del sitio sin que interfiera en el trabajo del programador. Esto hace posible que sean personas diferentes. Puede suceder que el programador realice cambios al contenido del sitio sin que esto afecte la estructura del diseño de las páginas.

Algunas de las características más significativas de Smarty se muestran a continuación:

- Es extremadamente rápido.

- Posee un elevado nivel de eficiencia.
- No analiza gramaticalmente desde arriba la plantilla, solo las compila una vez.
- Está atento para solo recompilar los archivos de plantilla que fueron cambiados.
- Se pueden crear funciones habituales y modificadoras de variables personalizados, de modo que el lenguaje de la plantilla es altamente extensible.
- Sintaxis de etiquetas delimitadoras para configuración de la plantilla, usándose {}, {{}}, <!--{}-->, etc.
- Permite un anidamiento ilimitado de section, if, etc. [10]

Selección del lenguaje y motor de plantilla a utilizar

Como lenguaje de programación del lado del servidor se ha decidido utilizar PHP que es un lenguaje interpretado por medio de script, creado fundamentalmente para agregar funcionalidades a servidores web. Además dispone de una gran cantidad de librerías que facilitan el trabajo del programador, incluye una extensión mejorada que permite explotar las capacidades de MySQL, puesto que es la base de datos por excelencia de este lenguaje. Es la que se utiliza con más frecuencia. Todo esto provoca que PHP sea un lenguaje robusto, rápido y fácil de aprender.

Este lenguaje posee como ventaja importante que es multiplataforma, puede ejecutarse tanto en Windows como en Linux, así como en otros sistemas operativos. Utilizando como motor de plantilla Smarty que permite que se diferencie y no se interfiera en el trabajo del programador y el diseñador. Además que se considera un motor de platilla rápido, eficiente y permite numerosas funciones.

Como lenguaje del lado del cliente se seleccionó JavaScript para crear pequeños programas con el propósito de ejecutar acciones dentro del contorno de la página web. Su uso se basa fundamentalmente en la creación de resultados específicos en las páginas, con alguna respuesta corta y rápida sin necesidad de crear una nueva página web para responder a determinada solicitud del cliente. Es compatible con la mayoría de los navegadores y tiene gran importancia en el desarrollo de páginas web. Es un lenguaje sencillo, pensado para realizar respuestas rápidas a las acciones de los usuarios, además de servir para la creación de contenidos dinámicos que cambien de color o movimiento.

Además de todas las características referenciadas sobre cada uno de los lenguajes de programación y plantillas, se ha decidido utilizar PHP, Smarty y JavaScript porque son del conocimiento de los desarrolladores de la aplicación y son los más difundidos en La Universidad de las Ciencias Informáticas.

1.4.3 Servidores web.

Un servidor web es un programa que se ejecuta en el servidor, escucha las peticiones HTTP que le llegan y las satisface. Dependiendo del tipo de la petición, el servidor web buscará una página web o bien ejecutará un programa en el servidor. El resultado producido por el servidor web es una página que se muestra en el navegador del cliente, el cual es el encargado de interpretar el código y mostrarlo de manera entendible al usuario.

Dos de los servidores web más conocidos y usados son el Apache y el IIS (Internet Information Server).

Apache.

Apache es un servidor web flexible, rápido y eficiente, continuamente actualizado y adaptado a los nuevos protocolos HTTP. Entre sus características destacan:

- Es un servidor web conforme al protocolo HTTP/1.1.
- Modular: Puede ser adaptado a diferentes entornos y necesidades, con los diferentes módulos de apoyo que proporciona, y con la API de programación de módulos, para el desarrollo de módulos específicos.
- Se desarrolla de forma abierta, cualquier persona con conocimientos de programación puede realizar aportes al mismo.
- Extensible: gracias a ser modular se han desarrollado diversas extensiones para los lenguajes de programación, entre las que destaca PHP.
- Multiplataforma: este servidor está disponible para varias plataformas, entre las que se destacan GNU/Linux, Windows, Mac OS y Mac OS X Server, entre otras.

1.4.4 Sistemas Gestores de Base de Datos.

Existen diferentes gestores de base de datos en la actualidad y que su funcionamiento es muy útil a la hora de implementar sistemas en los que se requiera almacenar información.

Los gestores de base de datos son un tipo de software que sirve de interfaz entre los usuarios, las aplicaciones y la base de datos. Utilizan lenguajes de manipulación de datos, de consultas y definiciones con el objetivo de manipular la información almacenada en ellos con facilidad y a gran velocidad.

SQL (Lenguaje de Consultas Estructuradas).

Este lenguaje está compuesto por sentencias, cláusulas, comandos y funciones que permiten conectar un sitio web a la base de datos, para realizar las búsquedas, actualizaciones, en fin, para manipular la información.

Es por lo tanto un estándar que consta de varias versiones (la última, del 2000). Nació como parte de un SGBD relacional denominado System R, a principios de los años 70. Evolucionó mucho desde aquellos tiempos y cambió su nombre desde SEQUEL hasta el SQL actual. Además consta de unas treinta sentencias. Cada sentencia demanda una acción específica por parte del SGBD tal como la creación de una nueva tabla, la recuperación de datos o la inserción de nuevos datos en la BD. [11]

Es un lenguaje normalizado que permite trabajar con diferentes lenguajes (ASP o PHP) aparejado con diferentes tipo de base de datos (MS Access, SQL Server, MySQL).

SQL admite una variada gama de tipos de datos para el tratamiento de la información contenida en las tablas, los tipos de datos pueden ser numéricos (con o sin decimales), alfanuméricos, de fecha o booleanos (si o no). Según el gestor de base de datos que se utilice los tipos de datos varían. SQL dispone de muy pocas órdenes, pero de múltiples palabras clave, lo que le convierten en un lenguaje sencillo pero tremendamente potente para llevar a cabo su función. [12]

El SQL presenta una gran desventaja es que solo puede utilizarse en aplicaciones sobre la plataforma Windows. Otra desventaja que presenta es que no es libre.

MySQL.

Es un gestor de base de datos multiusuarios que permite gestionar base de datos. Es muy aceptado por los usuarios porque trabaja bajo el concepto de código libre y también se puede utilizar cuando se necesite el trabajo de simultáneas sesiones.

MySQL es la base de datos Open Source más popular. Como base de datos relacional, utiliza múltiples tablas para almacenar y organizar la información. MySQL fue escrito en C y C++ y posee una gran adaptación a diferentes entornos de desarrollo, permitiendo su interacción con los lenguajes de programación más utilizados como PHP, Perl y Java y su integración en distintos sistemas operativos, entre ellos Windows y Linux. Se puede utilizar de forma gratuita e incluso permite modificar su código con

total libertad. Esto ha favorecido muy positivamente en su desarrollo y continuas actualizaciones, para hacer de MySQL una de las herramientas más utilizadas por los programadores orientados a Internet. [13]

El software de bases de datos MySQL consiste de un sistema cliente/servidor que se compone de un servidor SQL multihilo, varios programas clientes y bibliotecas, herramientas administrativas, y una gran variedad de interfaces de programación (APIs). Se puede obtener también como una biblioteca multihilo que se puede enlazar dentro de otras aplicaciones para obtener un producto más pequeño, más rápido, y más fácil de manejar. [14]

MySQL aprovecha la fortaleza de sistemas multiprocesador, gracias a su implementación multihilo, puede soportar grandes cantidades de tipos de datos para las columnas, hasta 32 índices por tabla. Posee gran portabilidad entre sistemas. Es evidente que el bajo consumo de este gestor de base de datos lo hacen apto para ser ejecutado en una máquina con pocos recursos. Presenta una probabilidad muy baja de corromper los datos.

PostgreSQL.

PostgreSQL es un gestor de base de datos orientado a objetos, presenta diferentes características como la herencia, funciones y restricciones. Soporta diferentes lenguajes como C, C++, Java, Python, PHP entre otros. Posee ciertas limitaciones técnicas como el no permitir consultas paralelas, lo que impide su adopción en un ambiente de trabajo de grandes dimensiones. Al usar el lenguaje de consulta SQL permite que se realicen trabajos en diferentes plataformas, además puede ser consultado directamente desde Internet.

La velocidad de respuesta que ofrece este gestor con bases de datos es relativamente pequeña, aunque esta misma velocidad la mantiene al gestionar bases de datos realmente grandes, cosa que resulta favorable. Por otra parte la primera interacción con dicho gestor puede provocar al usuario algún tipo de dificultad con la sintaxis de sus comandos.

Consume gran cantidad de recursos, y presenta un límite de 8K por fila, aunque se puede aumentar a 32K, con una disminución considerable del rendimiento, todo esto provoca que sea de 2 a 3 veces más lento que MySQL.

Selección del SGBD a utilizar.

Según la investigación realizada se ha determinado utilizar en el desarrollo del sistema como gestor de base de datos MySQL porque presenta alto rendimiento.

Además MySQL es un gestor de base de datos con una gran capacidad de respuesta y fiabilidad. Trabaja de manera rápida y eficiente porque almacena la información en diferentes tablas relacionadas, en vez de agruparlo todo en una sola, esto proporciona mayor venta y productividad a la hora de agregar, realizar consultas, lo que hace posible una mejor capacidad de respuesta. Es multiplataforma pues las aplicaciones que utilicen este gestor de base de datos para interactuar con la información del sitio web pueden realizarlo sobre Windows y Linux al mismo tiempo.

1.4.5 Estilo de Arquitectura.

La programación en tres capas es un estilo que permite separar la información en diferentes capas, capa de presentación, la capa de negocio y la capa de acceso a datos.

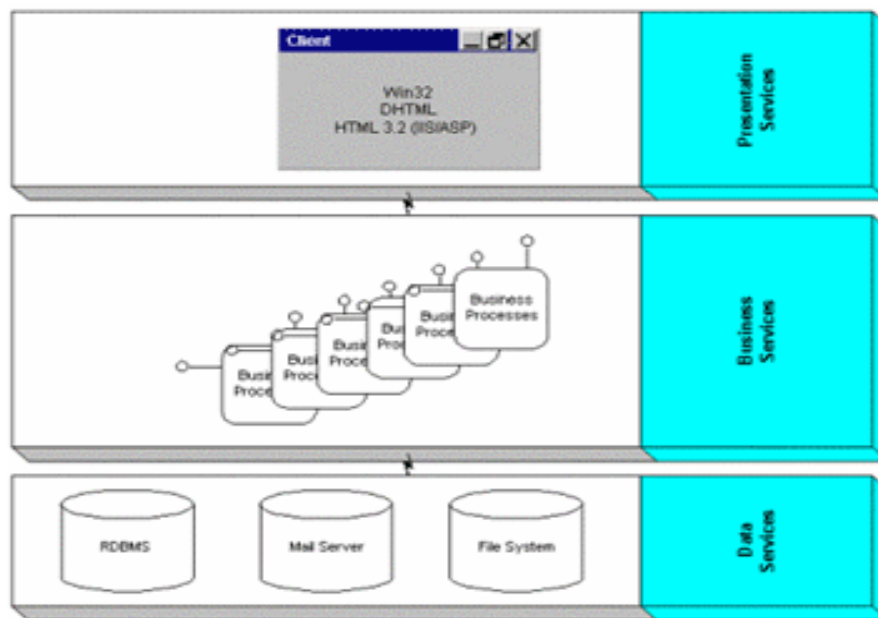


Figura. 1 Representación de la Arquitectura en 3 Capas.

Capa de presentación: es la capa que presenta el sistema al usuario, le comunica la información y captura los datos introducidos por el usuario (realiza un filtrado previo para comprobar que no hay errores de formato). Esta capa se comunica únicamente con la capa de negocio.

Capa de negocio: es la capa donde se implementa toda la lógica del negocio. Esta capa se comunica con la capa de presentación, para recibir las solicitudes y presentar los resultados, y con la capa de datos, para solicitar o recuperar los datos al gestor de base de datos.

Capa de datos: en esta capa residen todos los datos de la aplicación. Está formada por un gestor de bases de datos que realiza todo el almacenamiento de la información, recibe solicitudes de almacenamiento o recuperación de información desde la capa de negocio. [15]

Dentro de las principales ventajas que provee la arquitectura en tres capas se enumeran a continuación las más significativas:

- Los componentes de la aplicación pueden ser desarrollados en cualquier lenguaje de programación y estar esparcidos en múltiples servidores permitiendo una mayor escalabilidad.
- Los problemas de limitación para las conexiones a las bases de datos se minimizan ya que la base de datos solo es vista desde la capa intermedia y no desde todos los clientes. Las conexiones y los driver de las bases de datos no tienen que estar en los clientes.
- Los componentes de aplicación de la capa intermedia pueden ser asegurados centralmente usando una infraestructura común. Se pueden conceder o denegar los permisos componente a componente simplificando la administración. [16]

Se determina que la arquitectura en 3 capas es la más idónea para desarrollar la aplicación informática debido a las ventajas que presenta. Una de las ventajas mas importantes es que todas las funcionalidades se implementan en capas distintas, por ello, para realizar alguna modificación basta con ir directamente a la capa que se encarga de ello.

1.4.6 Herramientas utilizadas en la implementación de la aplicación.

Para el desarrollo de la aplicación se debe considerar una serie de herramientas que brindan una serie de ventajas. Se hace necesario el uso de editores, de un software para tratar las imágenes. Además de otras herramientas que resultan de utilidad.

Dreamweaver.

Dreamweaver es un editor HTML que proporciona el diseño, codificación y desarrollo de páginas, sitios, o aplicaciones web. Permite escribir manualmente el código HTML. Se pueden crear las páginas o sitios por medios de componentes visuales en la vista del diseño, sin necesidad de escribir ningún tipo de código y diseñar de forma rápida, enorme ventaja para las personas que desean diseñar y no saben mucho de programación.

Además los programadores por excelencia suelen utilizar Dreamweaver porque se puede trabajar a puro código ya que se incluyen numerosas herramientas y funciones que permiten realizar un diseño web. Esta herramienta utiliza lenguajes de programación como ASP.NET, Java, y PHP.

Dreamweaver MX 2004 proporciona un espacio de trabajo conformado por una sola ventana, donde se puede escoger entre un diseño orientado al diseñador y un diseño orientado a las necesidades de los usuarios que crean código manualmente.

Photoshop.

Es una herramienta profesional para la edición digital o procesamiento de imágenes. Se pueden realizar diseño web, fotografía y vídeo. Todas sus versiones realizadas a lo largo de la historia traen nuevos y mejorados servicios sólo con el propósito de optimizar la calidad y tratamiento de imágenes que favorezcan la perfección de los diseños, lo mismo para la web que restauraciones de gráficos.

Nusphere.

Este editor de PHP se caracteriza por:

- Integración de Plataforma, se integra con varias plataformas.
- Soporte para base de datos PostgreSQL, MySQL, Oracle y MSSQL.
- Soporta los lenguajes de programación PHP, HTML, CSS, JavaScript, XML, ASP, Perl, SQL, C/C++, y Python.
- Encuentra y reemplaza en el código en todos los archivos del proyecto o en cualquier parte de él.
- Proporciona instancias de sintaxis mientras se escriben códigos.
- Encuentra con facilidad y rapidez errores lógicos y de sintaxis.

Este editor permite al programador realizar la implementación de códigos de una manera rápida, y ayuda a encontrar posibles errores al redactarlos.

ERStudio.

Es una potente herramienta en el modelado, diseño y arquitectura de bases de datos. Permite que se puedan modelar y visualizar mejor las reglas de un negocio, además permite a los administradores de base de datos llevar un control para crear o mantener bases de datos. Permite modelar bases de datos para múltiples plataformas. ERStudio ayuda a agregar calidad a las base de datos pues refuerza los principios de diseño en la medida que se vaya modelando. Permite obtener diseños flexibles y con alto rendimiento.

Rational Rose Enterprise Edition.

Es la herramienta que permite construir y desarrollar a través del UML los casos de uso diferentes diagramas, modelando los flujos de trabajo por los que transita el desarrollo de un software. Permite la realización de los diferentes diagramas y la posterior generación del código, todo orientado a objetos. Posee librerías que facilitan la obtención de una ingeniería inversa sobre diferentes lenguajes como Java, C++, Corba, XML_DTD, ADA, Visual Basic.

Esta herramienta posibilita gestionar la evolución del ciclo de vida de un proyecto de software. Rational Rose aligera la implementación al automatizar los modelos arquitectónicos. Además permite visualizar, entender, y refinar los requerimientos y arquitectura antes de introducirse en el código. Establece una plataforma para automatizar arquitecturas de mejores prácticas hechas para soluciones tecnológicas específicas.

La Suite del Rational brinda una serie de productos que complementan su funcionamiento a la hora de implementar un software, ofreciendo diversas ventajas, como la administración de la configuración del software o de las solicitudes de cambios permitiendo que los administradores o jefes de proyecto puedan llevar un control sobre la evolución del proyecto de software que se implemente.

Rational Rose presenta características que todos los desarrolladores, analistas, y arquitectos exigen como son: un soporte UML incomparable, completo soporte al equipo, desarrollo basado en componentes, etc.[17]

1.5 Conclusiones.

En este capítulo se realizó una investigación sobre los flujos de trabajo. Se analizó el funcionamiento del flujo de trabajo de Gestión de Recursos Audiovisuales dentro de la DCAV. De los flujos de trabajo analizados se concluyó que el Flujo de Trabajo de Producción es el que se aplica al proceso de Gestión de Recursos Audiovisuales. Se analizaron las metodologías, lenguajes de programación, herramientas y sistemas gestores de bases de datos que pueden formar parte de la propuesta de solución.

2

CAPÍTULO Características del Sistema

2.1 Introducción.

En el presente capítulo se plantea el objeto de estudio, el objeto de automatización e información que se maneja en el proceso, además de la propuesta del sistema que se desea implementar.

Se realiza la modelación del negocio para determinar cuales son los actores y trabajadores que intervienen. Se especifican los casos de uso del negocio, describiéndose las actividades en las descripciones textuales. Se construyen los diagramas de casos de uso del negocio, clase de objetos y de actividades. También se identifican las reglas del negocio en las cuales se describen políticas o condiciones que deben cumplirse. Se realiza un análisis para definir los requisitos funcionales y no funcionales del sistema. Además se precisan y describen detalladamente los casos de uso del sistema. Se definen los actores y trabajadores.

2.2 Problema y situación problemática.

Para realizar las solicitudes de recursos audiovisuales se sigue un los Vicedecanos y Líderes de Proyecto redacten un documento con los datos de las solicitudes y los envían por correo electrónico a la DCAV. Una vez que llega la solicitud es revisada y si no contiene errores se envía a gestionar. En caso contrario se reenvía por correo electrónico a la persona que la realizó para que corrija los errores. Una vez que la solicitud es revisada se procede a gestionar los recursos audiovisuales de la misma. Después de gestionados se validan los recursos audiovisuales técnica y pedagógicamente y se notifica al cliente.

Las actividades llevadas a cabo para el flujo de trabajo de Gestión de Recursos Audiovisuales actualmente se realizan de forma manual. Existe un número reducido de personas que realizan estas actividades y un elevado volumen de información a procesar. Esto produce una serie de inconvenientes, puesto que existe una gran probabilidad de que los datos de las solicitudes se extravíen, se produce cúmulo de solicitudes ya que éstas personas no pueden laborar sobre todas las solicitudes al mismo tiempo. Se producen retrasos en las entregas de los recursos audiovisuales a los clientes debido a que

algunas solicitudes tienen un elevado número de recursos audiovisuales. No se tiene un control sobre en qué estado se encuentra cada recurso audiovisual durante alguna etapa del proceso de gestión.

2.3 Objeto de automatización.

Dentro de los procesos que se llevan a cabo en la DCAV, se desea automatizar el relacionado con La Gestión de Recursos Audiovisuales. El mismo se inicia cuando los Vicedecanos y Líderes de Proyecto realizan las solicitudes de recursos audiovisuales. Posteriormente envían la solicitud por correo electrónico a la DCAV, donde son validadas por especialistas de la dirección para eliminar los errores. Si la solicitud contiene errores se reenvía a la persona que realizó la solicitud informando los problemas para que sean corregidos. El Especialista puede definir una prioridad a la solicitud si el proyecto lo requiere.

Cuando la solicitud no contiene errores se procede a gestionar. Una vez gestionados los recursos audiovisuales de la solicitud el Especialista de la DCAV hace la entrega de los mismos y redacta La Planilla de Medias Entregadas, además de copiar los recursos audiovisuales en el servidor. Los recursos audiovisuales de la solicitud se validan técnicamente por los Líderes de Proyecto y pedagógicamente por el Especialista para garantizar que cumplen con los requerimientos de producción especificados en la solicitud. Si no contienen errores se procede a crear El Documento de Aceptación por parte del Especialista. Por el contrario, si contienen errores se crea un listado de recursos audiovisuales no aceptados, los cuales se envían a gestionar nuevamente.

En el Anexo 1 se muestra cada etapa por la que transita una solicitud realizada a la DCAV durante el proceso de gestión de recursos audiovisuales.[Anexo 1].

El Proceso de Gestión de Recursos Audiovisuales es sumamente importante en la creación de multimedia, por lo cual se requiere de su automatización de manera inmediata porque actualmente presenta numerosos inconvenientes que conducen a retrasos en las entregas a los clientes.

2.4 Información que se maneja.

Durante el flujo de trabajo de Gestión de Recursos Audiovisuales, el personal encargado de llevar a cabo todo el proceso manipula la información necesaria para gestionar cada recurso audiovisual que esté contenido en las solicitudes de los clientes. Cuando los clientes realizan una petición a la DCAV lo hacen a través de una solicitud donde se almacenan los datos del cliente, un listado de los recursos audiovisuales que desea gestionar y los requerimientos de producción. Además se encuentra cada

recurso audiovisual que constituye una información en sí, o sea, las imágenes, sonidos, locuciones, digitalizaciones, animaciones y videos. Por otro lado se encuentran los cronogramas de entregas en el cual se especifican los recursos audiovisuales que fueron encontrados en el DataCenter o en bases de datos externas o las que se deben producir y la planilla de entregada de recursos audiovisuales.

2.5 Propuesta del sistema.

Tendiendo en cuenta la problemática planteada, se propone el desarrollo de una aplicación web que permita tener un mayor control sobre las solicitudes de recursos audiovisuales, dando la posibilidad de conocer en qué estado se encuentra una recurso audiovisual del proceso. Dicha aplicación facilitará el trabajo con las solicitudes, puesto que todos los datos de los recursos audiovisuales se encontrarán en la solicitud almacenada en un mecanismo persistente.

Una de las ventajas de desarrollar una aplicación web radica en que el cliente no necesita instalar ningún componente adicional para interactuar con la misma, solo necesita un navegador.

El sistema permitirá informar rápidamente a los clientes cuando los recursos audiovisuales de su solicitud se encuentren gestionados.

Según lo anteriormente expuesto, se puede determinar que el soporte principal de la propuesta de solución será una aplicación web soportada por un servidor Apache que se desarrollará en el lenguaje de programación PHP. Se seleccionó MySQL como gestor por excelencia de este lenguaje por su ventaja de ser multiplataforma y por ser un SGBD de código abierto. También se escogió, para facilitar el trabajo el motor de plantillas Smarty.

Utilizando la metodología RUP para realizar el análisis, diseño implementación y prueba de sistema que se requiere implementar, con ayuda de las potentes herramientas que posee esta metodología: el paquete de Rational, el cual guiará el ciclo de vida del software y proporcionará un control sobre cada actividad y tarea destinada a realizar, pues Rational Rose unifica todo el equipo de desarrollo a través de la modelación, que se basa en UML.

La aplicación presentará una ayuda para suprimir cualquier duda que se les pueda presentar en determinado momento a los usuarios en su interacción con dicho sistema.

2.6 Modelo de Negocio.

Los objetivos fundamentales del proceso de modelación del negocio son:

- Adquirir una visión general de la organización.
- Identificar los problemas existentes y las mejoras potenciales.
- Lograr un entendimiento común entre los usuarios finales, clientes y desarrolladores.
- Derivar los requerimientos del sistema que va a soportar la organización.

2.6.1 Actores del Negocio.

Los actores del negocio son personas, organizaciones, sistemas automatizados o máquinas que interactúan externamente con el negocio y reciben algún beneficio.

En el negocio analizado se ha identificado como actor al Cliente:

Actor	Justificación
Cliente	Representa a las personas que realizan solicitudes. Obtienen como beneficio los recursos audiovisuales gestionados.

2.6.2 Trabajadores del Negocio.

Los trabajadores del negocio definen el comportamiento de individuos o sistemas automatizados que trabajan en colectivo realizando las actividades, manipulan entidades y no obtienen ningún resultado de valor. Representan roles.

Los trabajadores identificados en el proceso de negocio se representan a continuación:

Trabajador	Justificación
Especialista	Representa a la persona encargada de modificar la prioridad a una solicitud, según el proyecto que lo requiera.
Gestor	Representa a la persona encargada de revisar las solicitudes realizadas por los clientes y enviarlas a gestionar.
Validador Técnico	Representa a la persona encargada de validar técnicamente los recursos audiovisuales de la solicitud que ya se produjeron.
Validador Pedagógico	Representa a la persona encargada de validar pedagógicamente los recursos audiovisuales de la solicitud que ya se produjeron.

2.6.3 Diagrama de Casos de Uso del Negocio.

En la modelación del negocio se ha determinado un actor y un caso de uso en el que se refleja todo el flujo de eventos desde el momento en que el cliente lo inicia. A continuación se muestra el diagrama de casos de uso del negocio, donde se relaciona al Cliente (actor) con el caso de uso **Realizar Peticiones de Medias**.

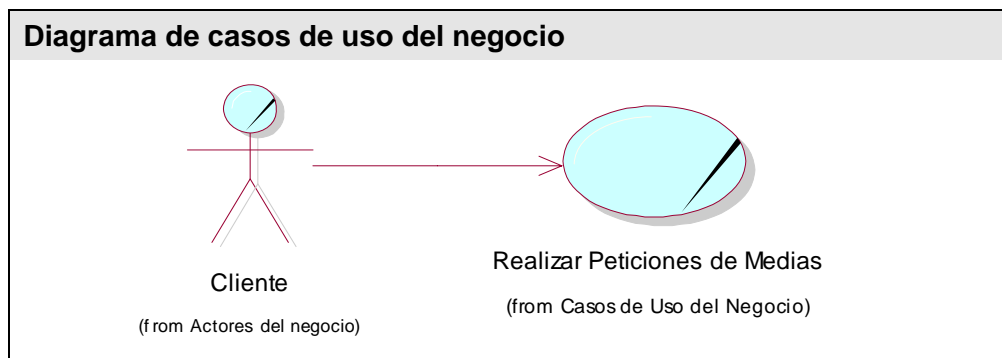


Figura. 2 Diagrama de Casos de Uso del Negocio.

2.6.4 Descripción del proceso de negocio.

A continuación se describe detalladamente como se realiza el proceso de negocio, el cual está definido por el caso de uso de negocio **Realizar Peticiones de medias**.

Caso de Uso	Realizar Peticiones de Medias	
Actores:	Cliente (Inicia)	
Propósito:	Obtener los recursos audiovisuales necesarios para realizar determinado producto.	
Resumen:	El Cliente redacta una solicitud de medias y la envía por correo a la DCAV, donde se revisa. Si la solicitud es aceptada se manda a gestionar. Cuando son gestionadas todas las medias de la solicitud se realiza una entrega de las medias. Se realiza la validación técnica y pedagógica verificando que cada media de la solicitud cumplan con los parámetros técnicos y pedagógicos establecidos. El caso de uso finaliza con el cierre del pedido emitiendo un documento de aceptación de las medias.	
Acciones del Actor	Respuestas del Negocio	
1. El caso de uso se inicia cuando el Cliente redacta la solicitud de media, el nombre de la planilla deberá estar conformado de la forma: Nombre.del.producto_Tipo.de.planilla_Número.de.solicitud y la envía por correo a la DCAV.	2. El Especialista modifica la prioridad de la solicitud, se especifica en la descripción: Prioridad Alta, definido por 'nombre de la persona' 'fecha'.	
	3. El Gestor revisa las solicitudes realizadas por los clientes a la DCAV. Define un responsable para la gestión de la solicitud.	
	4. El Gestor envía a gestionar las solicitudes.	
	5. El Especialista de la DCAV copia las medias en el servidor y modifica el estado de la planilla a gestionada, especificando en el documento la fecha de entrega.	
	6. Las medias gestionadas son validadas por el Validador Técnico y pedagógico, que verifican que cada media de la solicitud cumplan con los parámetros técnicos y pedagógicos establecidos.	
	7. Las medias de las solicitudes cumplen con los parámetros técnicos y pedagógicos establecidos, se cierra el pedido, emitiendo un documento de aceptación de las medias.	
8. El Cliente recoge las medias de su solicitud en un servidor.		
Cursos Alternos:		

<p>1.1 El Cliente corrige los datos de la solicitud que fue rechazada por la DCAV.</p>	
	<p>2.1 No se modifica la prioridad de la solicitud. La prioridad es la fecha de creada la solicitud.</p>
	<p>3.1 La solicitud contiene errores, se rechaza y se define en la descripción del documento el error: Rechazado por motivo 'tal' 'fecha'. Envía un correo al cliente con la solicitud rechazada para que corrija los errores.</p>
	<p>6.1 Las medias de la solicitud no cumplen con alguno de los parámetros técnicos y pedagógicos, se rechazan y se modifica el estado a la solicitud y a cada media a rechazadas, en las descripciones se especifican las razones por la que se rechazan.</p>
<p>Prioridad</p>	<p>Responde al problema de automatización en el proceso de Gestión de Medias, al darle solución a gran parte de ellos.</p>
<p>Mejoras</p>	<p>La automatización de realizar peticiones de medias facilitará el proceso de gestión de medias, aumentando la calidad y eficiencia, permitiendo un mejor control sobre las solicitudes de los clientes y ayudando a que los trabajadores realicen mejor su trabajo al gestionar las medias.</p>

2.6.5 Diagrama de Clases del Modelo de Objeto.

Los diagramas de clases de objetos se representan por entidades y trabajadores del negocio y las relaciones entre ellos. Las entidades del negocio representan un contenedor de información, algo físico que se utiliza en el proceso de negocio y que sirve para obtener, actualizar, o manejar información dentro del proceso de negocio y que a menudo es persistente. En la siguiente figura se representa el modelo de objetos del caso de uso del negocio **Realizar Peticiones de Medias**.

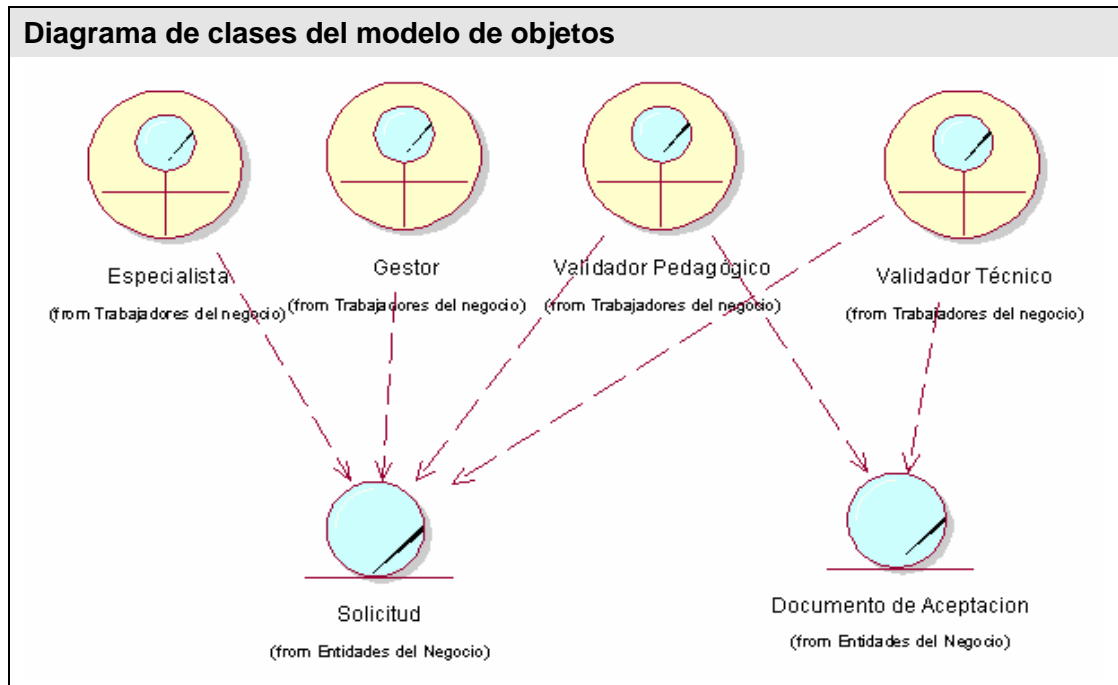


Figura. 3 Diagrama de Clases de Objetos.

2.6.6 Diagrama de Actividades.

Los diagramas de actividades son grafos en los que se especifican los estados por los que puede transitar una actividad, contienen bifurcaciones e hilos concurrentes.

Para realizar el diagrama de actividades del caso de uso **Realizar Peticiones de Medias** se han determinado diferentes actividades, como por ejemplo Redactar Solicitud, Definir Prioridad, Revisar Solicitud, Corregir Solicitud, Gestionar Medias y Validación Técnica y Pedagógica, además de seleccionar el actor y los trabajadores del negocio, en conjunto con la solicitud y el documento de aceptación como entidades del negocio determinadas con anterioridad. [Anexo 2].

2.6.7 Reglas del Negocio.

- Las solicitudes solo pueden ser realizadas por Vicedecanos y Jefes de Líneas.
- Cualquier persona involucrada en el proceso de gestión de la solicitud puede conocer el estado de las mismas.
- Los recursos audiovisuales de las solicitudes que no se pueden gestionar o producir se cancelan o se sustituyen.

- Las descripciones de las propiedades de los documentos no se pueden modificar en otra fecha después de agregada, aunque haya sido realizada por la misma persona.
- El cliente cuenta con un plazo de 7 días para recoger los recursos audiovisuales gestionados del servidor.
- Solo puede cancelar o sustituir un recurso audiovisual el cliente que realiza la solicitud.

2.7 Requisitos funcionales del sistema.

Después de conocer como se comporta el proceso de negocio se pueden identificar las funcionalidades que tendrá la aplicación a través de los requerimientos funcionales. Los requerimientos funcionales son capacidades o condiciones que el sistema debe cumplir.

El sistema debe permitir:

Gestión de solicitud

R1.Realizar solicitud de recursos audiovisuales (Nombre de la solicitud, Fecha Creada, Proyecto, Tipo de Solicitud, Solicitante).

R1.1 Insertar listado de requerimientos de producción a la solicitud según tipo de solicitud.

- Solicitud de animaciones: Formato, valores.
- Solicitud de digitalización: Formato, valores.
- Solicitud de sonido: Formato, resolución, frecuencia, tipo según canal.
- Solicitud de video: Formato, valores.
- Solicitud de locuciones: Formato, valores.
- Solicitud de imágenes: Formato, tamaño, tipo, resolución.

R1.2 Insertar listado de recursos audiovisuales a la solicitud según tipo de solicitud.

- Solicitud de animaciones: ID de la media, origen, descripción.
- Solicitud de digitalización: ID de la media, nombre requerido.
- Solicitud de sonido: ID de la media, descripción, tipo de sonido, origen, titulo, categoría.
- Solicitud de video: ID de la media, origen, descripción.
- Solicitud de locuciones: ID de la media, personajes que intervienen, texto, origen.
- Solicitud de imágenes: ID de la media, origen, descripción.

- R2. Almacenar la solicitud en un mecanismo persistente.
- R3. Buscar solicitudes.
- R4. Modificar el estado de cada recurso audiovisual de la solicitud.
- R5. Modificar la prioridad de las solicitudes.
- R6. Actualizar los datos de los recursos audiovisuales.
- R7. Imprimir la solicitud.
- R8. Almacenar en el historial los datos de las solicitudes terminadas.
- R9. Enviar notificación automática por correo electrónico.

Administración

- R10. Insertar nuevos usuarios en el sistema.
- R11. Eliminar usuarios inscritos en el sistema.
- R12. Actualizar el perfil de los usuarios.
- R13. Asignar un miembro del proyecto a la solicitud.
- R14. Restringir el acceso a las funcionalidades de acuerdo a los roles asignados a los actores del sistema.
- R15. Asignar roles a los usuarios.
- R16. Asignar proyecto a los usuarios.
- R17. Insertar Roles.
- R18. Modificar Roles.
- R19. Eliminar Roles.
- R20. Insertar páginas.
- R21. Modificar páginas.
- R22. Eliminar páginas.
- R23. Insertar Estados.

- R24. Modificar Estados.
- R25. Insertar Proyectos.
- R26. Modificar Proyectos.
- R27. Eliminar Proyectos.
- R28. Insertar Formatos.
- R29. Modificar Formatos.
- R30. Eliminar Formatos.
- R31. Cambiar la contraseña.

Gestión de Información

- R32. Generar Reportes Personalizados.
- R33. Imprimir Reporte.

2.8 Requerimientos no Funcionales.

Los requerimientos no funcionales son propiedades o cualidades que el sistema debe tener, son características que hagan al producto usable, rápido o confiable para los clientes finales.

Apariencia, interfaz externa:

- El sistema posee una interfaz amigable y bien definida.
- Puede ser utilizado sin previa preparación de los usuarios debido a la sencillez de las interfaces.
- Identificación de colores y formatos con acciones del sistema. Los mensajes de error devueltos desde el servidor se muestran en color rojo, mientras que los relacionados con acciones satisfactorias se muestran en color negro.

Usabilidad:

- Para utilizar el sistema no es necesario poseer conocimientos elementales de computación. El sistema podrá ser usado por cualquier persona.

Seguridad:

Confidencialidad:

- Se emplean sesiones para restringir el acceso a las páginas.
- Los usuarios pueden cambiar su contraseña.
- Los usuarios se autentican para poder acceder al contenido del sistema, según los roles que tienen asignados.
- Las contraseñas se envían encriptadas desde la página del cliente hacia las páginas servidoras utilizando el algoritmo de cifrado MD5. Se almacenan cifradas en la base de datos, de forma que solo el usuario conoce su contraseña.

Integridad:

- Se garantiza la integridad de la información del sistema porque sólo personas autorizadas podrán realizar cambios.
- Se solicita confirmación para las operaciones irreversibles, tales como eliminaciones en la base de datos.

Disponibilidad:

- El sistema estará disponible las 24 horas del día, todos los días del año.

Portabilidad:

- El sistema desarrollado es multiplataforma.

Confiabilidad:

Entre las múltiples características que hacen al sistema confiable se encuentran:

- Uso de transacciones.
- Copias de seguridad.

Funcionalidad:

- Los procesos de búsquedas se realizan de manera rápida.
- Posee una navegación fácil para el usuario.

Rendimiento:

- Los tiempos de acceso a la base de datos son relativamente pequeños, no excediendo 5 segundos.
- El acceso vía web a la aplicación se efectúa de forma rápida.

Legales:

- La aplicación desarrollada no viola ninguna ley establecida en La Constitución de la República de Cuba.

Ayuda y documentación en línea:

- El sistema posee una ayuda online para que el usuario en cualquier momento pueda consultarla en caso de encontrar alguna duda durante su trabajo con la aplicación.

Software:

- Navegador compatible con Internet Explorer 6 o superior, Mozilla Firefox.

Restricciones en el diseño y la implementación.

- El sistema debe ser una aplicación web.
- Se debe desarrollar utilizando lenguajes de programación libres.

2.9 Definición de los Casos de Uso.

2.9.1 Actores del Sistema.

Los actores del sistema son terceros fuera del sistema que interactúan con él.

Actores del sistema	Justificación
Administrador	Representa a la persona encargada de realizar las tareas de administración tales como insertar usuarios en el sistema, eliminarlos, actualizar sus datos, gestionar los roles, etc.
Cliente	Representa a la persona que utiliza el sistema para redactar una solicitud de recursos audiovisuales.
Gestor	Representa a la persona que valida las solicitudes para garantizar que están bien redactadas, también envía a gestionar las solicitudes que se aceptan.

Validador	Generaliza los roles de Validador Técnico y Validador Pedagógico.
Validador Pedagógico	Representa a la persona que valida pedagógicamente los recursos audiovisuales de la solicitud.
Validador Técnico	Representa a la persona que valida técnicamente los recursos audiovisuales de la solicitud.
Invitado	Representa a la persona que utiliza el sistema para consultar los reportes.
Especialista	Representa a la persona que establece la prioridad a las solicitudes de recursos audiovisuales.

2.9.2 Diagrama de Casos de Uso del Sistema.

Los casos de uso del sistema son fracciones de la funcionalidad de un sistema, describen la sucesión de determinados eventos que realiza un actor al interactuar con la aplicación y representan los requerimientos funcionales del sistema.

Para lograr una mejor organización del trabajo se agrupó la mayor cantidad de casos de usos por paquetes. El criterio seguido para realizar la agrupación fue para dar soporte a un determinado actor del sistema. Como resultado se crearon 3 paquetes:

Paquete Administración: En este paquete se encuentran los casos de uso relacionados con la administración del sistema, casos de uso que sólo son realizados por el Administrador.

Paquete Gestión de Solicitudes: En este paquete se agrupan los casos de uso que son realizados por el Cliente y el Especialista.

Paquete Gestión de Medias: En este paquete se agrupan los casos de uso realizados por el Gestor y los Validadores (Validador Técnico y Pedagógico).

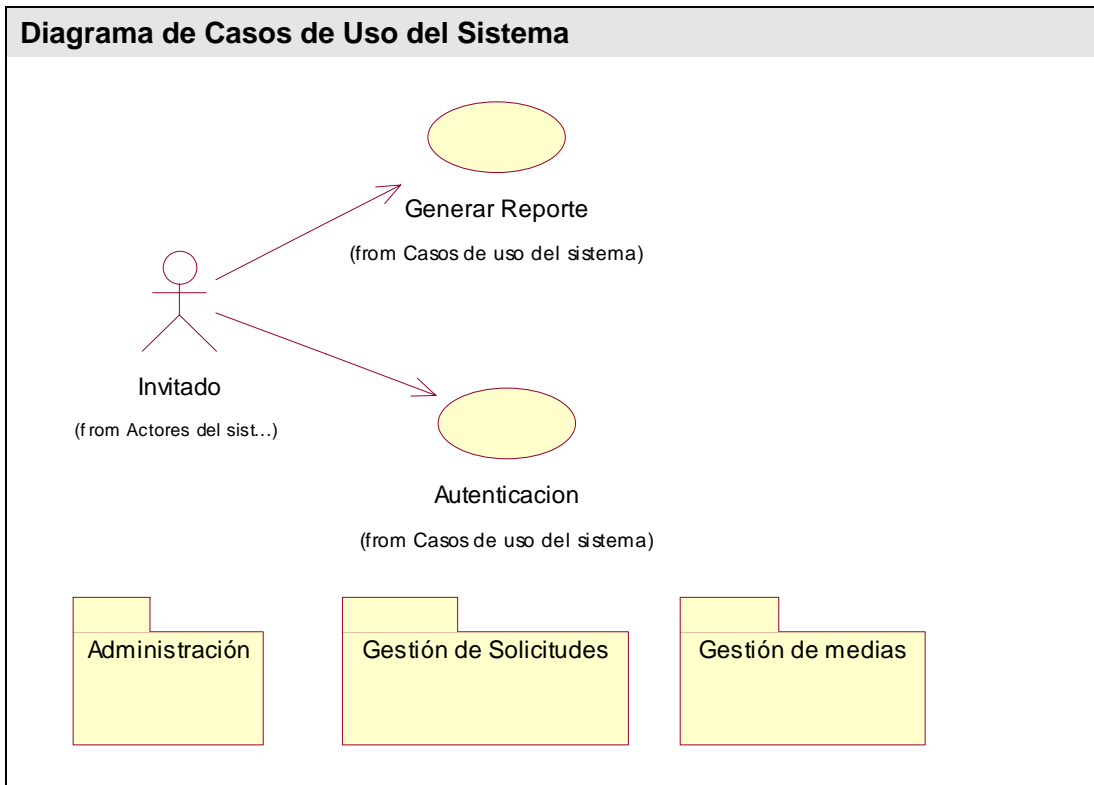


Figura. 4 Diagrama General de Casos de Uso del Sistema.

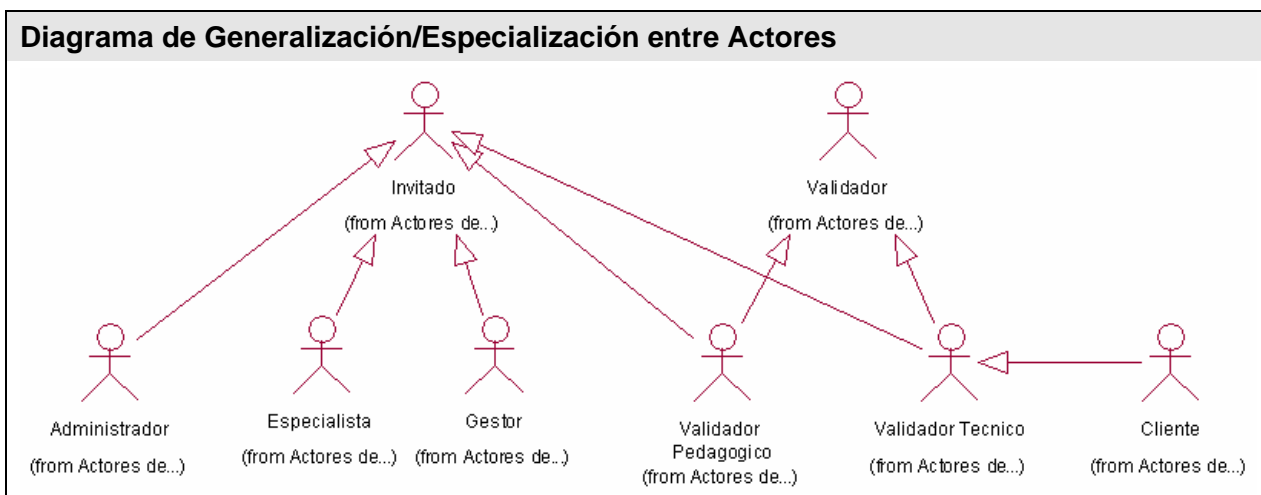


Figura. 5 Diagrama de Generalización/Especialización entre Actores.

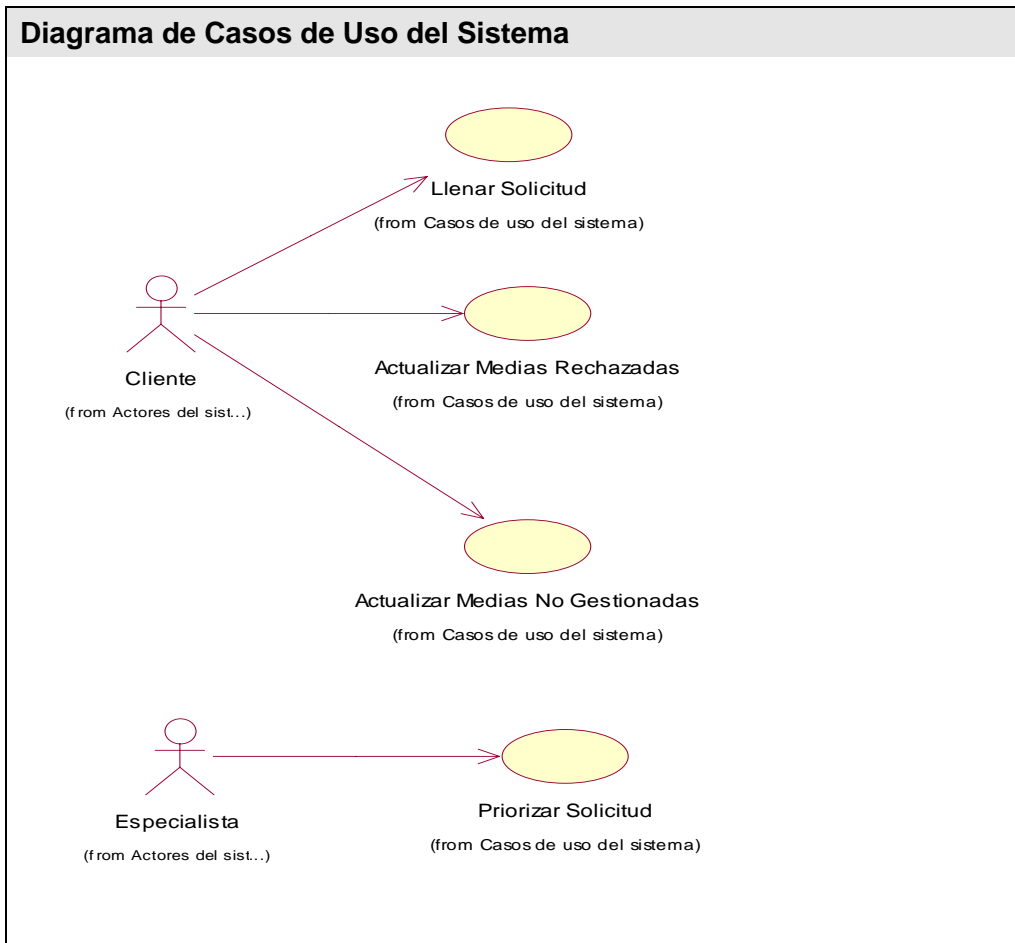


Figura. 6 Diagrama de Casos de Uso. Paquete Gestión de Solicitudes.

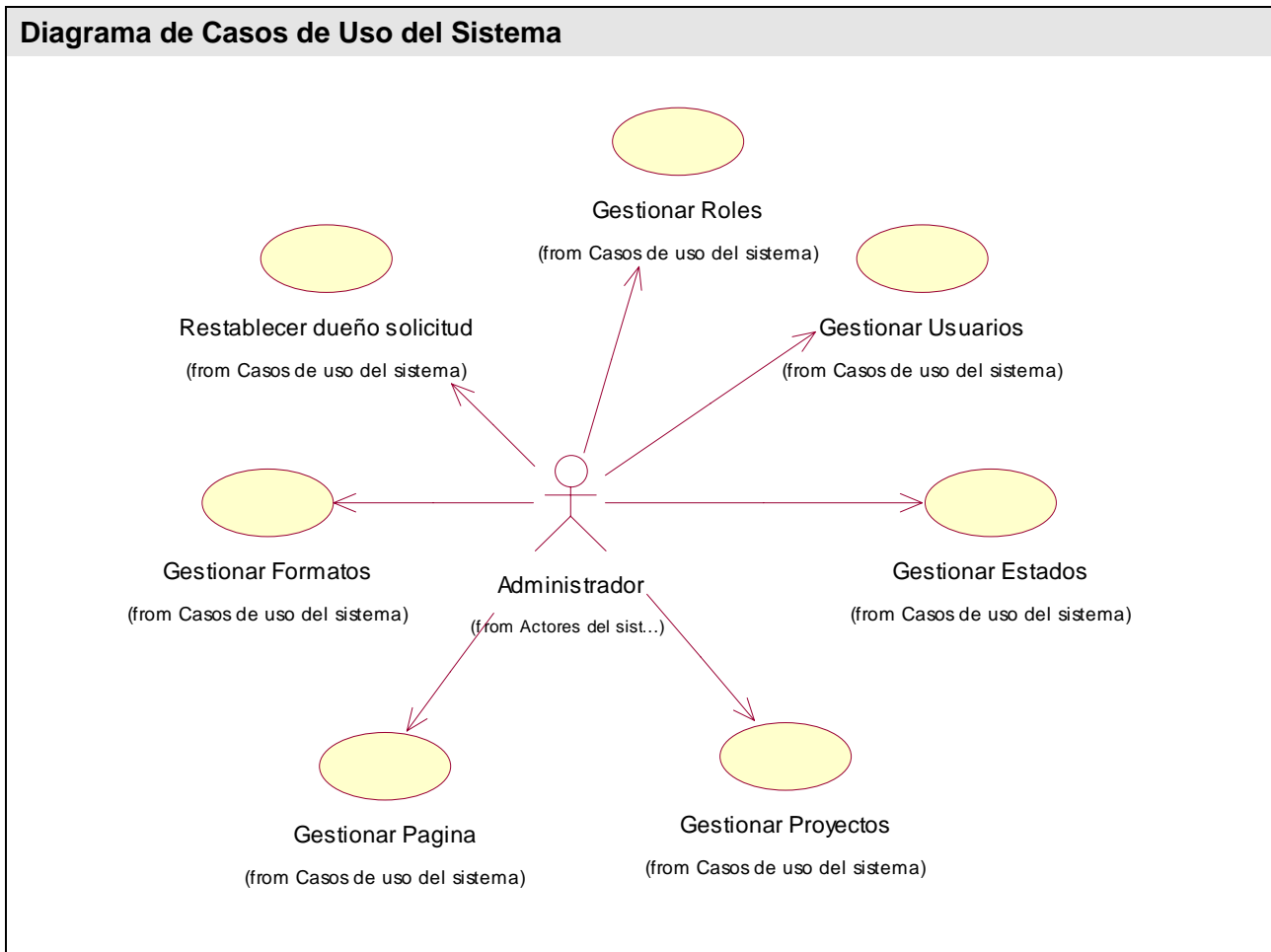


Figura. 7 Diagrama de Casos de Uso. Paquete Administración.

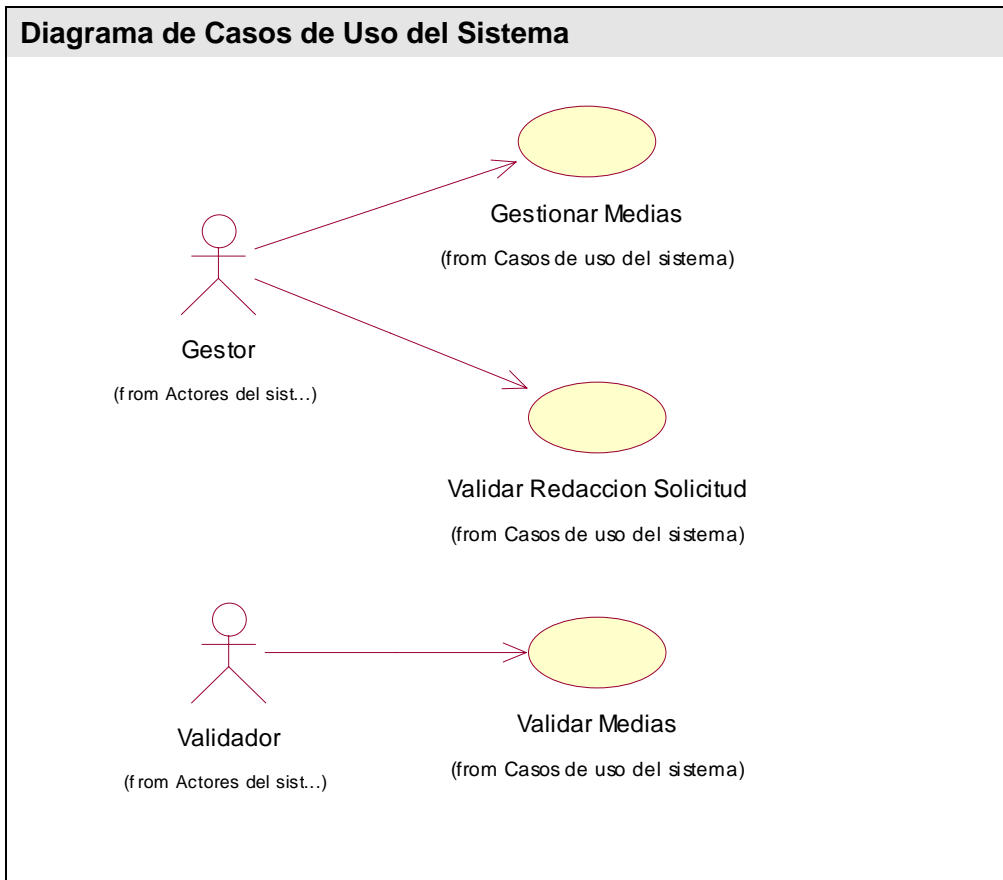


Figura. 8 Diagrama de Casos de Uso. Paquete Gestión de Medias.

2.9.3 Descripción textual de los Casos de Uso del Sistema.

En las descripciones textuales se describen a un nivel detallado los eventos que se realizan en cada caso de uso. A partir de este punto se comienza a construir el sistema. A continuación se presentan las descripciones breves de los casos de uso del sistema.

Caso de Uso	Autenticación
Actores:	Invitado (Inicia)
Prioridad:	Secundario
Resumen:	Autenticar los usuarios para obtener acceso a las funcionalidades de la aplicación.
Referencia:	R14, R31.

Caso de Uso	Generar Reportes
Actores:	Invitado (Inicia)
Prioridad:	Secundario
Resumen:	Generar reportes personalizados según las opciones que hayan sido seleccionadas para realizar el mismo.
Referencia:	R32, R33.

Caso de Uso	Gestionar Usuarios
Actores:	Administrador (Inicia)
Prioridad:	Secundario
Resumen:	Insertar nuevos usuarios en el sistema si no existen previamente. También modificar, eliminar, asignar roles y proyectos a los usuarios que se encuentren registrados en la aplicación.
Referencia:	R10, R11, R12, R15, R16.

Caso de Uso	Gestionar Roles
Actores:	Administrador (Inicia)
Prioridad:	Secundario
Resumen:	El Administrador crea, modifica, elimina los roles y crea los vínculos a dichos roles (Gestor, Cliente, Validador Técnico, Validador Pedagógico, Especialista, Administrador) según sea definido por la administración.
Referencia:	R17, R18, R19.

Caso de Uso	Gestionar Página
Actores:	Administrador (Inicia)
Prioridad:	Secundario
Resumen:	El Administrador inserta, modifica, elimina las páginas que se les asignan a los roles del sistema.
Referencia:	R20, R21, R22.

Caso de Uso	Gestionar Formatos
Actores:	Administrador (Inicia)
Prioridad:	Secundario
Resumen:	El Administrador inserta, modifica y elimina los formatos de las medias de las solicitudes.
Referencia:	R28, R29, R30.

Caso de Uso	Gestionar Proyectos
Actores:	Administrador (Inicia)
Prioridad:	Secundario
Resumen:	Insertar nuevos proyectos, modificar y eliminar los proyectos existentes en la base de datos.
Referencia:	R25, R26, R27.

Caso de Uso	Gestionar Estados
Actores:	Administrador (Inicia)
Prioridad:	Secundario
Resumen:	El Administrador inserta, modifica y elimina los estados por lo que transitan las medias y las solicitudes(Creada, Aceptada, Rechazada, Gestionada, Cancelada, Pendiente, Terminada, No Gestionada).
Referencia:	R23, R24.

Caso de Uso	Restablecer Dueño Solicitud
Actores:	Administrador (Inicia)
Prioridad:	Secundario
Resumen:	Cambiar el responsable de la solicitud si se elimina de la base de datos la persona que había realizado la misma.
Referencia:	R3, R13.

Caso de Uso	Llenar Solicitud
Actores:	Cliente(Inicia)
Prioridad:	Crítico
Resumen:	Insertar nuevas solicitudes de medias en la base de datos.
Referencia:	R1, R1.1, R1.2, R2, R9.

Caso de Uso	Actualizar Medias Rechazadas
Actores:	Cliente (Inicia)
Prioridad:	Secundario
Resumen:	Actualizar los datos de las medias que tienen problemas de redacción.
Referencia:	R4, R6, R9.

Caso de Uso	Actualizar Medias No Gestionadas
Actores:	Cliente (Inicia)
Prioridad:	Secundario
Resumen:	Modificar las características de las medias que no se pueden producir ni encontrar en el DataCenter. También se pueden cancelar las medias si ya no se desean producir.
Referencia:	R4, R6, R9.

Caso de Uso	Validar Redacción Solicitud
Actores:	Gestor (Inicia)
Prioridad:	Crítico
Resumen:	Validar la redacción de las solicitudes de medias.
Referencia:	R4, R9.

Caso de Uso	Priorizar Solicitud
Actores:	Especialista (Inicia)
Prioridad:	Secundario
Resumen:	Modificar la prioridad de alguna solicitud si el proyecto lo requiere.
Referencia:	R5, R9.

Caso de Uso	Gestionar Medias
Actores:	Gestor (Inicia)
Prioridad:	Crítico
Resumen:	Modificar el estado de las medias que se producen.
Referencia:	R4, R7, R9.

Caso de Uso	Validar Medias
Actores:	Validador (Inicia)
Prioridad:	Crítico
Resumen:	Verificar que las medias de las solicitudes que ya están producidas, cumplan con los parámetros técnicos y pedagógicos establecidos.
Referencia:	R4, R8, R9.

Las siguientes descripciones detalladas corresponden a los casos de uso críticos del sistema. Las restantes descripciones detalladas se muestran en los anexos. [Anexo 3].

Caso de Uso	Llenar Solicitud
Actores:	Cliente (Inicia)
Propósito:	Llenar una solicitud de media.
Resumen:	Los clientes insertan nuevas solicitudes de medias en la base de datos para los proyectos productivos a los cuales pertenecen.
Referencias:	R1, R1.1, R1.2, R2, R9.
Precondiciones:	El Cliente se autentica en el sistema.
Poscondiciones:	Se almacena la nueva solicitud en la base de datos.

CU Asociados:	
Curso Normal de los Eventos	
Acciones de Actor	Respuesta del Sistema
<p>1. El caso de uso se inicia cuando el Cliente selecciona la opción “Llenar Solicitud”.</p>	<p>2. Muestra un formulario para introducir los datos de la solicitud. Se muestra un campo desplegable con los tipos de solicitudes y otro con los proyectos asociados al solicitante.</p> <p>Los datos introducidos en las solicitudes son: Nombre de la solicitud, Solicitante, Tipo de Solicitud, Proyecto.</p> <p>Los datos introducidos en la lista de requerimientos de producción para cada solicitud son:</p> <p>Solicitud de animaciones. (Formato, valores)</p> <p>Solicitud de digitalización. (Formato, valores)</p> <p>Solicitud de sonido. (Formato, resolución, frecuencia, tipo según canal)</p> <p>Solicitud de video. (Formato, tamaño)</p> <p>Solicitud de locuciones. (Formato, valores)</p> <p>Solicitud de imágenes. (Formato, tamaño, tipo, resolución)</p> <p>Los datos introducidos en la lista de medias para cada solicitud son:</p> <p>Solicitud de animaciones. (ID, origen, descripción)</p> <p>Solicitud de digitalización. (ID, nombre requerido)</p> <p>Solicitud de sonido. (ID, descripción, tipo de sonido, origen, titulo, categoría, autor)</p> <p>Solicitud de video. (ID, origen, descripción)</p> <p>Solicitud de locuciones. (ID, personajes que intervienen, texto, origen)</p> <p>Solicitud de imágenes. (ID, origen, descripción)</p>

<p>3. Introduce los datos de la solicitud en la "Pantalla 1" y oprime sobre el botón "Continuar".</p>	<p>4. Almacena temporalmente los datos de la solicitud y muestra el formulario para introducir los requerimientos de la solicitud.</p>
<p>5. Introduce los datos de los requerimientos en la "Pantalla 2" y oprime sobre el botón "Continuar".</p>	<p>6. Almacena temporalmente los requerimientos y muestra el formulario para introducir las medias de la solicitud.</p>
<p>7. Introduce los datos de las medias de la solicitud en la "Pantalla 3" y oprime sobre el botón "Añadir Elementos".</p>	<p>8. Almacena temporalmente la media y muestra el formulario para introducir nuevas medias. Se repite el Paso 9.</p>

Pantalla 1

Planilla de Nueva Solicitud :

Nombre de la solicitud: *

Solicitante: *

Tipo de solicitud: *

Proyecto: *

Continuar->

Pantalla 2

Requerimientos de Producción de Animaciones:

Formato: mpg avi Cinepack DivX5

Valores: * * * *

[Nuevo Formato](#)

Continuar->

Pantalla 3

Listado de medias de la solicitud:

Id:	Descripción:	Origen:
A01	* Movimientos de aprobación de la mascota, cara sonriente.	* La animación se encuentra en el sitio http://www.cubasi.cu

Añadir Elementos **Vista Previa** **Terminar**

Pantalla 4

Introduzca el nuevo formato para Animaciones:

Formato: *

Insertar

Pantalla 5

Planilla de Solicitud de Animaciones:

Nombre de la solicitud:	lgg	Fecha:	2007-05-29
Solicitante:	Esneiker Enriquez Cabrera	Tipo de solicitud:	Animaciones
Proyecto:	PDVS	Prioridad:	Normal

Requerimientos de Producción:

Formato:	Valores:
<input type="checkbox"/> swf	800x600

Listado de medias de la solicitud:

Id:	Descripción:	Origen:
<input type="checkbox"/> A1	Movimientos de aprobación de la mascota, cara sonriente; en el caso de responder correctamente.	La animación se encuentra en el sitio http://www.cubasi.cu
<input type="checkbox"/> A2	Gesto de tristeza de la mascota, si se equivoca parcialmente o no logra descubrir la imagen.	

<-Atrás

Eliminar Marcados

Terminar

Curso Alterno: Agregar nuevo formato.

5.1 El Cliente hace clic sobre el vínculo "Nuevo Formato" en la "Pantalla 2".	5.2 Muestra un formulario para introducir los datos del nuevo formato.
5.3 Introduce los datos del nuevo formato en la "Pantalla 4" y oprime sobre el botón "Insertar".	5.4 Almacena el nuevo formato en la base de datos.

Curso Alterno: Se oprime el botón "Vista Previa".

7.1 El Cliente oprime sobre el botón "Vista Previa" en la "Pantalla 3".	7.2 Muestra los datos introducidos en la solicitud en la "Pantalla 5".
7.3 El Cliente selecciona la(s) media(a) o el(los) requerimiento(s) que desea eliminar de la solicitud y oprime sobre el botón "Eliminar Marcados" en la "Pantalla 3".	7.4 Elimina los elementos seleccionados.

Curso Alterno: Insertar la solicitud.

7.1.1 El Cliente oprime sobre el botón	7.1.2 Almacena la solicitud en la base de datos, envía un correo
---	---

"Terminar".		al Gestor notificando que existen nuevas solicitudes, finalizando el caso de uso.
Curso Alterno: Excede el límite de medias por solicitud.		
7.1.a El Cliente excede el límite de medias establecidas por solicitud.		7.1.b Muestra un mensaje notificando que ha superado el límite de medias para esa solicitud, las restantes debe ponerlas en una nueva solicitud.
Prioridad:		Crítico
Caso de Uso	Validar Redacción Solicitud	
Actores:	Gestor(Inicia)	
Propósito:	Validar los datos de las solicitudes realizadas.	
Resumen:	Validar la redacción de las solicitudes realizadas para garantizar que no contengan errores.	
Referencias:	R4, R9.	
Precondiciones:	El Gestor se autentica en el sistema.	
Poscondiciones:	Se validan las medias de la solicitud y se cambia el estado de las medias validadas a "Aceptada" o "Rechazada".	
CU Asociados:		
Curso Normal de los Eventos		
Acciones de Actor		Respuesta del Sistema
1. El caso de uso se inicia cuando el Gestor selecciona la opción "Validar Redacción de Solicitudes".		2. Muestra un listado de solicitudes agrupadas por prioridad. Listando las solicitudes nuevas y que se están validando separadas de las solicitudes que han sido arregladas por los clientes y la fecha de realización de las solicitudes.
3. Selecciona el nombre de una solicitud en la "Pantalla 1".		4. Muestra los datos de la solicitud en la "Pantalla 2".
5. Valida los datos de las medias de la solicitud. No contienen errores y cambia el estado de cada media de la solicitud a "Aceptada", especifica la fecha en la que estarán listas las medias de la solicitud. Oprime sobre el botón "Actualizar Solicitud" en la "Pantalla 2".		6. Actualiza el estado de las medias de la solicitud y almacena la descripción, la fecha de gestión y el nombre del responsable de la gestión, finalizando el caso de uso.
Pantalla 1		

Listado de nuevas solicitudes y solicitudes que se están revisando:	
Solicitudes con prioridad Alta	Fecha de realizada la solicitud:
Solicitudes con prioridad Normal	Fecha de realizada la solicitud:
A jugar A S2	2007-05-29
lqq A S4	2007-05-29

Pantalla 2

Planilla de Solicitud de Animaciones:

Nombre de la solicitud: new_A_54 Fecha: 2007-06-18
 Solicitante: Esneiker Enriquez Cabrera Tipo de solicitud: Animaciones
 Proyecto: PDVS Prioridad: Normal

Requerimientos de Producción:

Formato: Valores:
 mpg sdf

Listado de medias de la solicitud:

Id:	Descripción:	Estado:	Observaciones:
<input checked="" type="checkbox"/> A01	sdf	Aceptada <input type="button" value="v"/> <input type="button" value="v"/>	<input type="text"/> <input type="button" value="v"/> <input type="button" value="v"/>

[Detalles](#)

Observaciones de la Solicitud:
 Fecha de Compromiso:(aaaa-mm-dd) Responsable:

Curso Alterno: Los datos de las medias de la solicitud contienen errores.

5.1 Se rechazan las medias que contienen errores, se cambia el estado de la media a "Rechazada", y se define en la descripción de la media el error.	5.2 Envía un correo al cliente correspondiente a la solicitud notificando el rechazo de la(s) media(s) de su solicitud. Actualiza el estado de la(s) medias (s) y almacena la descripción en la base de datos.
Prioridad:	Crítico

Caso de Uso	Gestionar Medias																					
Actores:	Gestor (Inicia).																					
Propósito:	Gestionar todas las medias de las solicitudes validadas satisfactoriamente.																					
Resumen:	Cambiar el estado de las medias producidas y las medias que no se pueden producir ni se encuentran en el DataCenter.																					
Referencias:	R4, R7, R9.																					
Precondiciones:	El Gestor se autentica en el sistema.																					
Poscondiciones:	Se gestionan las medias de la solicitud.																					
CU Asociados:																						
Curso Normal de los Eventos																						
Acciones de Actor	Respuesta del Sistema																					
1. El caso de uso se inicia cuando el Gestor selecciona la opción "Gestionar Medias".	2. Lista el nombre de las solicitudes que tienen medias "Aceptada" y "Pendiente" y la fecha en que fueron realizadas, agrupadas por prioridad.																					
3. Selecciona el nombre de una solicitud en la "Pantalla 1".	4. Muestra los datos de la solicitud.																					
5. Cambia el estado de las medias producidas a "Gestionada", oprime sobre el botón "Actualizar Solicitud" en la "Pantalla 2".	6. Almacena el estado de las medias en la base de datos, envía un correo a los Validadores Técnicos y Pedagógicos que revisan las solicitudes del proyecto al cual pertenece la solicitud que se gestionó. Finaliza el caso de uso.																					
Pantalla 1																						
<p>Listado de solicitudes con prioridad Alta:</p> <table border="1"> <thead> <tr> <th>Nombre de las solicitudes:</th> <th>Fecha de realizada la solicitud:</th> </tr> </thead> <tbody> <tr> <td>anidmsd A 52</td> <td>2007-05-28</td> </tr> <tr> <td>sqfs A 53</td> <td>2007-05-28</td> </tr> <tr> <td>asfsdf A 51</td> <td>2007-05-29</td> </tr> <tr> <td>A jugar A 52</td> <td>2007-05-29</td> </tr> </tbody> </table> <p>Listado de solicitudes con prioridad Normal:</p> <table border="1"> <thead> <tr> <th>Nombre de la solicitud:</th> <th>Fecha de realizada la solicitud:</th> </tr> </thead> <tbody> <tr> <td>anidmsd A 52</td> <td>2007-05-28</td> </tr> <tr> <td>sqfs A 53</td> <td>2007-05-28</td> </tr> <tr> <td>asfsdf A 51</td> <td>2007-05-29</td> </tr> <tr> <td>A jugar A 52</td> <td>2007-05-29</td> </tr> </tbody> </table>			Nombre de las solicitudes:	Fecha de realizada la solicitud:	anidmsd A 52	2007-05-28	sqfs A 53	2007-05-28	asfsdf A 51	2007-05-29	A jugar A 52	2007-05-29	Nombre de la solicitud:	Fecha de realizada la solicitud:	anidmsd A 52	2007-05-28	sqfs A 53	2007-05-28	asfsdf A 51	2007-05-29	A jugar A 52	2007-05-29
Nombre de las solicitudes:	Fecha de realizada la solicitud:																					
anidmsd A 52	2007-05-28																					
sqfs A 53	2007-05-28																					
asfsdf A 51	2007-05-29																					
A jugar A 52	2007-05-29																					
Nombre de la solicitud:	Fecha de realizada la solicitud:																					
anidmsd A 52	2007-05-28																					
sqfs A 53	2007-05-28																					
asfsdf A 51	2007-05-29																					
A jugar A 52	2007-05-29																					
Pantalla 2																						

Planilla de Solicitud de Animaciones:

Nombre de la solicitud: new_A_54 Fecha: 2007-06-18
 Solicitante: Esneiker Enriquez Cabrera Tipo de solicitud: Animaciones
 Proyecto: PDVS Prioridad: Normal

Requerimientos de Producción:

Formato: Valores:
 mpg sdf

Listado de medias de la solicitud:

Id:	Descripción:	Estado:	Observaciones:	
<input checked="" type="checkbox"/> A01	sdf	Gestionada		Detalles

Pantalla 3

Listado de solicitudes Rechazadas con prioridad Alta:

Nombre de las solicitudes: **Fecha de creada:**

Listado de solicitudes Rechazadas con prioridad Normal:

Nombre de la solicitud: **Fecha de creada:**

[A jugar A_52](#) 2007-05-29

Pantalla 4

Planilla de Solicitud de Animaciones:

Nombre de la solicitud: A jugar_A_52 Fecha: 2007-05-29
 Solicitante: Esneiker Enriquez Cabrera Tipo de solicitud: Animaciones
 Proyecto: SIS Prioridad: Normal

Requerimientos de Producción:

Formato: Valores:
 swf 800x600

Medias con problemas Técnicos y Pedagógicos:

Id:	Estado:	Problemas Pedagógicos:	Problemas Técnicos:	Observaciones:	
<input checked="" type="checkbox"/> A1	Gestionada	La media no cumple con los parámetros pedagógicos establecidos.	La media tiene problemas..		Detalles

Curso Alterno: No se encuentran las medias en el DataCenter.

5.1 Oprime sobre el botón “Imprimir” en la “Pantalla 2”.

5.2 Imprime la solicitud en la impresora.

Curso Alterno: Gestionar solicitudes con medias rechazadas por la validación.	
1.1.1 El Gestor selecciona la opción "Gestionar Medias".	1.1.2 Lista el nombre de las solicitudes que tienen medias "Aceptada" y "Pendiente", la fecha en que fueron realizadas, agrupadas por prioridad.
1.1.3 Selecciona una solicitud en la "Pantalla 3".	1.1.4 Muestra los datos de la solicitud, todos los requerimientos y se listan solamente los datos de las medias que tienen estado "Pendiente".
1.1.5 Gestiona la media, cambia el estado y oprime sobre el botón "Actualizar Solicitud" en la "Pantalla 4".	1.1.6 Almacena el estado de la media en la base de datos.
1.1.7 Envía la solicitud a gestionar.	
Curso Alterno: No se puede gestionar las medias.	
1.5.1 No se puede gestionar las medias de la solicitud, se cambia el estado a las medias a "No Gestionada" en la pantalla 2 y oprime sobre el botón "Actualizar Solicitud".	1.5.2 Actualiza el estado de las medias de la solicitud en la base de datos y envía un correo al cliente de dicha solicitud notificando que las medias correspondientes a la misma no se pudieron gestionar ni producir.
Prioridad:	Crítico

Caso de Uso	Validar Medias
Actores:	Validador (Inicia).
Propósito:	Validar técnica y pedagógicamente las medias gestionadas y almacenar las solicitudes en el historial.
Resumen:	El Validador verifica que las medias de las solicitudes que ya están gestionadas, cumplan con los parámetros técnicos y pedagógicos establecidos.
Referencias:	R4, R8, R9.
Precondiciones:	El Validador se autentica en el sistema y las medias de la solicitud están gestionadas.
Poscondiciones:	Se aceptan las medias de la solicitud y se notifica al cliente.
CU Asociados:	
Curso Normal de los Eventos	
Acciones de Actor	Respuesta del Sistema
1. El caso de uso se inicia cuando el Validador selecciona la opción "Validación Técnica".	2. Lista el nombre de la solicitud que tiene estado "Gestionada" y que no tengan dicha validación, agrupadas por prioridad, la fecha de realización de la misma.

<p>3. Selecciona una solicitud en la "Pantalla 1".</p>	<p>4. Muestra los datos de la solicitud.</p>
<p>5. Valida las medias correspondientes a la solicitud. Las medias cumplen todos los parámetros técnicos establecidos en los requerimientos de la solicitud, selecciona en el campo de aceptación de cada media "Si". Oprime sobre el botón "Actualizar Solicitud" en la "Pantalla 2".</p>	<p>6. Registra el nombre de usuario de la persona que valida las medias y la fecha de validación. Comprueba que todas las medias de la solicitud han sido validadas técnica y pedagógicamente. Si el campo de aceptación de cada media, en cada validación es "Si" establece el estado de la solicitud a "Terminada" y envía un correo al cliente y a todos los involucrados en el proceso de gestión de dicha solicitud notificando que ya la solicitud se encuentra terminada. Se informa al cliente que tiene un plazo de 7 días para recoger las medias del servidor.</p>
	<p>7. Se almacenan los datos de la solicitud en el historial y se eliminan los datos de la base de datos, finalizando el caso de uso.</p>

Pantalla 1

Listado de solicitudes con prioridad Alta:

Nombre de la solicitud:

Fecha de realización:

Listado de solicitudes con prioridad Normal:

Nombre de la solicitud:

Fecha de realización:

[A Jugar 2 A S3](#)

2007-05-21

Pantalla 2

Planilla de Solicitud de Animaciones:

Nombre de la solicitud:	A jugar_A_S2	Fecha:	2007-05-29
Solicitante:	Esneider Enriquez Cabrera	Tipo de solicitud:	Animaciones
Proyecto:	SIS	Prioridad:	Normal

Requerimientos de Producción:

Formato:	Valores:
swf	800x600

Listado de medias de la solicitud:

Id:	Descripción:	Aceptación:	Observaciones:	
<input checked="" type="checkbox"/> A1	Movimientos de aprobación de la mascota, cara sonriente; en el caso de responder correctamente.	SI <input type="button" value="v"/>	<input type="text"/>	Detalles

Actualizar Solicitud

Pantalla 3

Listado de solicitudes con prioridad Alta:	
Nombre de la solicitud:	Fecha de realización:
Listado de solicitudes con prioridad Normal:	
Nombre de la solicitud:	Fecha de realización:
A Jugar 2 A S3	2007-05-21

Pantalla 4

Planilla de Solicitud de Animaciones:

Nombre de la solicitud: A jugar_A_S2 Fecha: 2007-05-29
 Solicitante: Esneiker Enriquez Cabrera Tipo de solicitud: Animaciones
 Proyecto: SIS Prioridad: Normal

Requerimientos de Producción:

Formato: Valores:
 swf 800x600

Listado de medias de la solicitud:

Id:	Descripción:	Aceptación:	Observaciones:	
<input checked="" type="checkbox"/> A1	Movimientos de aprobación de la mascota, cara sonriente; en el caso de responder correctamente.	<input type="text" value="Si"/>	<input type="text"/>	Detalles

Sección: Validación Pedagógica.	
1.1 Selecciona la opción “Validación Pedagógica”.	1.2 Lista el nombre de la solicitud que tiene estado gestionada y que no tengan dicha validación, agrupadas por prioridad, la fecha de realización de la misma.
1.3 Selecciona una solicitud en la “Pantalla 3”.	1.4 Muestra los datos de la solicitud.
1.5 Valida las medias correspondientes a la solicitud. Las medias cumplen todos los parámetros pedagógicos establecidos en los requerimientos de la solicitud. Selecciona en el campo de aceptación de cada media “Si”. El campo de las “Observaciones” se muestra inhabilitado. Oprime sobre el botón “Actualizar Solicitud” en la “Pantalla 4”.	1.6 Registra el nombre de usuario de la persona que valida las medias y la fecha de validación. Comprueba que todas las medias de la solicitud han sido validadas técnica y pedagógicamente. Si el campo de aceptación de cada media, en cada validación es “Si” establece el estado de la solicitud a “Terminada” y envía un correo al cliente y a todos los involucrados en el proceso de gestión de dicha solicitud notificando que ya la solicitud se encuentra terminada. Se informa al cliente que tiene un plazo de 7 días para recoger las medias del servidor. 1.7 Se almacenan los datos de la solicitud en el historial y se eliminan los datos de la base de datos.
Curso alterno: La media no cumple con los parámetros pedagógicos establecidos.	
1.5.1 Las medias no cumplen los parámetros pedagógicos establecidos,	1.5.2 Registra el nombre de usuario de la persona que valida las medias y la fecha de validación. Comprueba que todas las

<p>selecciona en el campo de aceptación "No". Introduce las "Observaciones", especificando cual fue el error de la media y oprime sobre el botón "Actualizar" en la "Pantalla 4".</p>	<p>medias de la solicitud han sido validadas pedagógicamente. Si el campo de aceptación de alguna media en la validación es "No" establece el estado de la media a "Pendiente", la medias no pasa el proceso de validación técnica y envía un correo al gestor notificando que existen medias pendientes de gestión.</p>
<p>Curso alterno: La media no cumple con los parámetros técnicos establecidos.</p>	
<p>5.1 Selecciona en el campo de aceptación "No". Introduce las "Observaciones", especificando cual fue el error de la media y oprime sobre el botón "Actualizar" en la "Pantalla 2".</p>	<p>5.2 Registra el nombre de usuario de la persona que valida las medias y la fecha de validación. Comprueba que todas las medias de la solicitud han sido validadas técnica y pedagógicamente. Si el campo de aceptación de alguna media en alguna validación es "No", establece el estado de las medias de la solicitud a "Pendiente" y envía un correo al gestor notificando que existen medias pendientes de gestión.</p>
<p>Prioridad:</p>	<p>Crítico</p>

2.10 Conclusiones.

En este capítulo se realizó el estudio de los problemas existentes en el proceso de gestión de recursos audiovisuales, se definió el objeto de automatización y la información que se maneja. Se realizó la propuesta del sistema para dar solución a los problemas existentes en el flujo de trabajo de Gestión de Recursos Audiovisuales en la UCI.

Se definieron los actores, trabajadores, casos de uso y entidades del negocio, representando sus relaciones en los diagramas de casos de uso del negocio, modelo de objeto y actividades, además de los actores y casos de uso del sistema. Se realizó la descripción textual del caso de uso del negocio y las descripciones de los casos de uso del sistema, detallando las diferentes actividades que se llevan a cabo para realizar una solicitud de recursos audiovisuales a la DCAV. Además se determinaron los requisitos funcionales y no funcionales que permitirán al sistema ser funcional.

3

CAPÍTULO Análisis y Diseño del Sistema

3.1 Introducción.

En este capítulo se realiza el análisis y diseño del sistema a través de la metodología seleccionada luego del estudio preliminar efectuado, RUP. Se desarrollaran los diagramas de clases del análisis, clases web, y diagramas de interacción. Se plantean los principios de diseño, los estándares de interfaz de aplicación y codificación. Además se desarrolla el modelo de base de datos, la cual soportará todas las acciones sobre la aplicación por los usuarios finales. Se realiza la concepción general de ayuda y tratamiento de errores.

3.2 Modelo de Análisis.

El análisis se fundamenta en la obtención de la visión del sistema, de investigar *qué* debe hacer a través de los requisitos funcionales ya que realiza un análisis detallado de los mismos. El objetivo que se persigue es la comprensión y claridad de las funcionalidades que deberá poseer el sistema para que cumpla con las expectativas de los usuarios finales al que está dirigido.

Entre los artefactos que generan los trabajadores de este flujo de trabajo se encuentran el modelo del análisis, clases del análisis, realización de los casos de uso entre otros.

Luego de realizar un análisis de los casos de uso del sistema se puede obtener el modelo de análisis, el cual proporciona una visión general del sistema, permite dividir el diseño y la implementación del sistema en pequeños módulos que facilitan el manejo y comprensión para el trabajo del equipo de desarrollo a la hora de implementar el software.

Los diagramas de clase se elaboran a través de la notación UML. Se emplean las clases del tipo interfaz, controladora y entidades, además de las asociaciones existentes entre ellas como son la asociación, composición y la multiplicidad o cardinalidad que pueden tener.

3.3 Modelo de Diseño.

El diseño como muestra de un refinamiento del análisis tiene presente los requisitos no funcionales, se enfoca en *cómo* responde el sistema a sus objetivos. Permite alcanzar una comprensión adecuada de los aspectos relacionados con las restricciones relacionadas con los lenguajes de programación, componentes reutilizables, sistemas operativos y tecnologías.

El desarrollo de un diseño preciso y a gran escala proporciona que el sistema sea implementado sin ambigüedades. Entre los artefactos que se generan se encuentran el modelo del diseño, descripción de la arquitectura, clases del diseño, entre otras.

3.3.1 Clases Persistentes.

Las clases persistentes son aquellas que tienen la capacidad de mantener su valor en el espacio y en el tiempo. Estas clases son la que se incluyen el modelo de objeto, del cual se obtiene el modelo de datos. Las clases persistentes relacionadas con el sistema de Gestión de Solicitudes de Recursos Audiovisuales se muestran en el diagrama de clases persistentes que se localiza en los anexos. [Anexo 4].

3.3.2 Modelo de Datos.

En el modelo de datos se relacionan las tablas mediante relaciones de asociación, agregación, composición, restricciones y la cardinalidad. Esto proporciona un mejor entendimiento para construir una base de datos que soporte las funcionalidades del sistema que se requiere desarrollar. [Anexo 5].

A continuación se muestran las tablas que conforman la base de datos perteneciente al sistema de Gestión de Solicitudes de Recursos Audiovisuales, con sus atributos y la descripción correspondiente a cada una.

Nombre: tb_usuario		
Descripción: En esta tabla se registran los datos que pertenecen a los usuarios.		
Atributo	Tipo	Descripción
nombre_usuario	Varchar(20)	Identificador de cada usuario.
contraseña	Text	La contraseña del usuario para iniciar sesión.
nombre	Varchar(50)	Nombre del usuario
apellidos	Varchar(100)	Apellidos del usuario
correo	Varchar(50)	Dirección de correo electrónico del usuario.
teléfono	Varchar(18)	Teléfono del usuario.

Nombre: tb_rol		
Descripción: Esta tabla contendrá los datos de los roles existentes en la base de datos.		
Atributo	Tipo	Descripción
id_rol	Integer	Identificador del rol.
nombre_rol	Varchar(30)	Nombre del rol.
descripción	Text	Breve descripción del rol.

Nombre: tb_pagina		
Descripción: Esta tabla contendrá los datos relacionados con las páginas se que registren y se asigne a un determinado rol.		
Atributo	Tipo	Descripción
id_pagina	Integer	Identificador de la página.
id_rol	Integer	Identificador del rol que relaciona con la página.
nombre	Varchar(60)	Nombre de la página.
dirección	Text	Dirección física de la página.

Nombre: tb_proyecto		
Descripción: En esta tabla se registran los datos de los proyectos.		
Atributo	Tipo	Descripción
id_proyecto	Integer	Identificador del proyecto.
proyecto	Varchar(100)	Nombre del proyecto.
entidad	Varchar(50)	Nombre de la entidad a la cual pertenece el proyecto.

Nombre: tb_solicitud_priorizada		
Descripción: Se registra la prioridad de la solicitud.		
Atributo	Tipo	Descripción
nombre_usuario	Varchar(20)	Usuario de la persona que prioriza la solicitud.
id_solicitud	Integer	Identificador de la solicitud priorizada.
descripcion_prioridad	Text	Motivo por el que se prioriza la solicitud.
fecha_prioridad	Date	Fecha en la que se prioriza la solicitud.

Nombre: tb_solicitud		
Descripción: En esta tabla se registran los datos de las solicitudes realizadas por los clientes.		
Atributo	Tipo	Descripción
id_solicitud	Integer	Identificador de la solicitud.
nombre_usuario	Varchar(20)	Identificador del usuario que realiza la solicitud.
id_estado	Integer	Identificador del estado de la solicitud.
nombre_solicitud	Varchar(100)	El nombre de la solicitud.
fecha_creada	Date	Fecha en que se crea la solicitud.
tipo	Varchar(20)	Tipo de solicitud que se puede realizar (imagen, sonido, locución, video, digitalización y animación).
id_proyecto	Integer	Identificador del proyecto al cual pertenece la solicitud.
descripcion_solicitud	Text	Motivo por el que se rechaza la solicitud en la validación de redacción.

fecha_gestion	Date	Fecha en que se gestiona completamente la solicitud.
fecha_compromiso	Date	Fecha en que terminará la gestión de la solicitud.
responsable_gestion	Varchar(100)	Nombre de la persona encargada de gestionar la solicitud.

Nombre: tb_estado		
Descripción: Esta tabla almacena los tipos de estados que pueden tener la solicitud y las medias.		
Atributo	Tipo	Descripción
id_estado	Integer	Identificador del estado.
tipo_estado	Varchar(20)	Nombre del estado.
descripción	Text	Breve descripción del estado

Nombre: tb_revision_media		
Descripción: Se registran los datos relacionados con la validación de la redacción de la media de la solicitud.		
Atributo	Tipo	Descripción
id_revision	Integer	Identificador de la revisión de media.
id_media	Integer	Identificador de la media que se revisa.
nombre_usuario	Varchar(20)	Usuario de la persona que realiza la revisión de la media.
fecha_revision	Date	Fecha en que se revisa la media.
descripcion_revision	Text	Motivo por el cual no se rechaza la media.

Nombre: tb_gestion_media		
Descripción: Se registran los datos relacionados con la gestión de la media.		
Atributo	Tipo	Descripción
id_gestion	Integer	Identificador de gestión.
id_media	Integer	Identificador de la media que se gestiona.
nombre_usuario	Varchar(20)	Usuario de la persona que realiza la gestión de la media.
fecha_gestion	Date	Fecha en que se gestiona la media.
descripcion_gestion	Text	Motivo por el cual no se puede gestionar la media.

Nombre: tb_revisión_técnica		
Descripción: Se registran los datos relacionados con la revisión técnica de la media.		
Atributo	Tipo	Descripción
id_revision	Integer	Identificador de la revisión técnica.
id_media	Integer	Identificador de la media que se valida técnicamente.
nombre_usuario	Varchar(20)	Usuario de la persona que realiza la validación técnica de la media.
aceptación	Varchar(10)	Aceptación de la media (Si ó No).
fecha_revision	Date	Fecha en que se valida técnicamente la media.
descripcion_revision	Text	Motivo por le cual no se acepta técnicamente la media.

Nombre: tb_revisión_pedagógica		
Descripción: Se registran los datos relacionados con la revisión pedagógica de la media.		
Atributo	Tipo	Descripción
id_revision	Integer	Identificador de la revisión pedagógica.
id_media	Integer	Identificador de la media que se valida pedagógicamente.
nombre_usuario	Varchar(20)	Usuario de la persona que realiza la validación pedagógica de la media.
aceptación	Varchar(10)	Aceptación de la media (Si ó No).
fecha_revision	Date	Fecha en que se valida pedagógicamente la media.
descripcion_revision	Text	Motivo por le cual no se acepta pedagógicamente la media.

Nombre: tb_formato		
Descripción: En esta tabla se almacenan los tipos de formatos que pueden tener los requerimientos de producción.		
Atributo	Tipo	Descripción
id_formato	Integer	Identificador del formato.
nombre_formato	Varchar(18)	Nombre del formato.
tipo_req	Varchar(20)	Requerimiento al que pertenece el formato (animación, digitalización, videos, locución imagen y sonido).

Nombre: tb_req_producción		
Descripción: Se almacenan los datos de los diferentes tipos de requerimientos de producción que puede tener la solicitud.		
Atributo	Tipo	Descripción
id_req	Integer	Identificador de requerimiento de producción.
id_formato	Integer	Identificador del formato al cual pertenece el requerimiento de producción.
id_solicitud	Integer	Identificador de la solicitud que tiene el requerimiento de producción.

Nombre: tb_r_anim_loc_digit		
Descripción: Se registran los datos relacionados con los requerimientos de producción de los siguientes tipos de solicitudes(Animación, Locución y Digitalización)		
Atributo	Tipo	Descripción
id_req	Integer	Identificador de requerimiento de producción.
valores	Varchar(20)	Valores

Nombre: tb_r_imágenes		
Descripción: Se registran los datos relacionados con los requerimientos de producción de solicitudes de imágenes.		
Atributo	Tipo	Descripción
id_req	Integer	Identificador de requerimiento de producción.
resolución	Varchar(20)	Resolución de la imagen.
tipo_imagen	Varchar(20)	Tipo de imagen.
tamaño	Varchar(20)	Tamaño de la imagen.

Nombre: tb_r_video		
Descripción: Se registran los datos relacionados con los requerimientos de producción de solicitudes de video.		
Atributo	Tipo	Descripción
id_req	Integer	Identificador de requerimiento de producción.
tamaño	Varchar(20)	Tamaño del video.

Nombre: tb_r_sonido		
Descripción: Se registran los datos relacionados con los requerimientos de producción de solicitudes de sonido.		
Atributo	Tipo	Descripción
id_req	Integer	Identificador de requerimiento de producción.
resolución	Varchar(20)	Resolución del sonido.
frecuencia	Varchar(20)	Frecuencia del sonido.
tipo_según_canal	Varchar(40)	Tipo de sonido según el canal.

Nombre: tb_media		
Descripción: Se almacenan los datos de la media de la solicitud.		
Atributo	Tipo	Descripción
id_media	Integer(auto incrementado)	Identificador de la media.
identificador	Varchar(18)	Identificador que se le asigna a la media.
id_solicitud	Integer	Identificador de la solicitud a la cual pertenece la media.
id_estado	Integer	Estado de la media.

Nombre: tb_m_anim_img_video		
Descripción: Se almacenan los datos relacionados con la media de la solicitud según el tipo que sea (Animación, Imagen o Video).		
Atributo	Tipo	Descripción
id_media	Integer	Identificador de la media.
origen	Text	Ubicación de la media (animación, imagen, video).
descripción	Text	Breve descripción de la media.

Nombre: tb_m_digitalización		
Descripción: Se almacenan los datos relacionados con la media de la solicitud del tipo "Digitalización".		
Atributo	Tipo	Descripción
id_media	Integer	Identificador de la media.
nombre_requerido	Varchar(50)	Nombre requerido de la digitalización.

Nombre: tb_m_locuciones		
Descripción: Se almacenan los datos relacionados con la media de la solicitud del tipo "Locución".		
Atributo	Tipo	Descripción
id_media	Integer	Identificador de la media.
origen	Text	Ubicación de la media (locución).
texto	Text	El texto de la locución.
personajes	Text	Nombre de los personajes que intervienen en la locución.

Nombre: tb_m_sonido		
Descripción: Se almacenan los datos relacionados con la media de la solicitud del tipo "Sonido".		
Atributo	Tipo	Descripción
id_media	Integer	Identificador de la media.
descripción	Text	Breve descripción de la media.
tipo_sonido	Varchar(30)	Tipo de sonido (Efecto sonoro, Música).
titulo	Varchar(30)	El título del sonido.
categoría	Varchar(20)	Tipo de categorías (Infantil, No Infantil, Instrumental y Cantada)
origen	Text	Ubicación de la media (sonido).
autor	Varchar(50)	Nombre del autor del sonido.

Nombre: tb_solicitud_h		
Descripción: Se almacenan los datos de la solicitud, una vez que se termina completamente.		
Atributo	Tipo	Descripción
id_solicitud	Integer	Identificador de la solicitud.
nombre_cliente	Varchar(100)	Nombre y Apellidos de la persona que creó la solicitud.
nombre_solicitud	Varchar(100)	El nombre de la solicitud.
fecha_creada	Date	Fecha en que se crea la solicitud.
tipo	Varchar(20)	Tipo de solicitud que se puede realizar (imagen, sonido, locución, video, digitalización y animación).
proyecto	Varchar(100)	Proyecto al cual pertenece la solicitud.

descripcion_solicitud		Motivo por el que se rechaza la solicitud en la validación de redacción.
fecha_gestion	Date	Fecha en que se gestionó completamente la solicitud.
fecha_compromiso	Date	Fecha en que terminará la gestión de la solicitud.
responsable_gestion	Varchar(100())	Nombre de la persona que gestionó la solicitud.

Nombre: tb_req_producción_h		
Descripción: Se almacenan los datos de requerimiento de producción de la solicitud, una vez que se termina completamente.		
Atributo	Tipo	Descripción
id_req	Integer	Identificador de requerimiento de producción.
id_formato	Integer	Identificador del formato al cual pertenece el requerimiento de producción.
id_solicitud	Integer	Identificador de la solicitud que tiene el requerimiento de producción.

Nombre: tb_r_anim_loc_digit_h		
Descripción: Se almacenan los datos de los requerimientos de producción de la solicitud (Animación, Locución, Digitalización), una vez que se termina completamente.		
Atributo	Tipo	Descripción
id_req	Integer	Identificador de requerimiento de producción.
valores	Varchar(20)	Valores

Nombre: tb_r_imágenes_h		
Descripción: Se almacenan los datos de los requerimientos de producción de la solicitud "Imagen".		
Atributo	Tipo	Descripción
id_req	Integer	Identificador de requerimiento de producción.
resolución	Varchar(20)	Resolución de la imagen.
tipo_imagen	Varchar(20)	Tipo de imagen.
tamaño	Varchar(20)	Tamaño de la imagen.

Nombre: tb_r_video_h		
Descripción: Se almacenan los datos de los requerimientos de producción de la solicitud "Video".		
Atributo	Tipo	Descripción
id_req	Integer	Identificador de requerimiento de producción.
tamaño	Varchar(20)	Tamaño del video.

Nombre: (tb_r_sonido_h		
Descripción: Se almacenan los datos de los requerimientos de producción de la solicitud "Sonido".		
Atributo	Tipo	Descripción
id_req	Integer	Identificador de requerimiento de producción.
resolución	Varchar(20)	Resolución del sonido.
frecuencia	Varchar(20)	Frecuencia del sonido.
tipo_según_canal	Varchar(40)	Tipo de sonido según el canal.

Nombre: tb_media_h		
Descripción: Se almacenan los datos de la media de la solicitud		
Atributo	Tipo	Descripción
id_media	Integer	Identificador de la media.
identificador	Varchar(18)	Identificador que se le asigna a la media.
id_solicitud	Integer	Identificador de la solicitud a la cual pertenece la media.

Nombre: tb_m_anim_img_video_h		
Descripción: Se almacenan los datos de la media de la solicitud del tipo "Animación, Imagen, Video".		
Atributo	Tipo	Descripción
id_media	Integer	Identificador de la media.
origen	Text	Ubicación de la media (animación, imagen, video).
descripción	Text	Breve descripción de la media.

Nombre: tb_m_digitalización_h		
Descripción: Se almacenan los datos de la media de la solicitud de Digitalización.		
Atributo	Tipo	Descripción
id_media	Integer	Identificador de la media.
nombre_requerido	Varchar(50)	Nombre requerido de la digitalización.

Nombre: tb_m_locuciones_h		
Descripción: Se almacenan los datos de la media de la solicitud de Locución.		
Atributo	Tipo	Descripción
id_media	Integer	Identificador de la media.
origen	Text	Ubicación de la media (locución).
texto	Text	El texto de la locución.
personajes	Text	Nombre de los personajes que intervienen en la locución.

Nombre: tb_m_sonido_h		
Descripción: Se almacenan los datos de la media de la solicitud de Sonido.		
Atributo	Tipo	Descripción
id_media	Integer	Identificador de la media.
descripción	Text	Breve descripción de la media.
tipo_sonido	Varchar(30)	Tipo de sonido (Efecto sonoro, Música).
titulo	Varchar(30)	El título del sonido.
categoría	Varchar(20)	Tipo de categorías (Infantil, No Infantil, Instrumental y Cantada).
origen	Text	Ubicación de la media (sonido).
autor	Varchar(50)	Nombre del autor del sonido.

3.3.3 Diagramas de Secuencia.

En los diagramas de secuencia se muestran las interacciones, en una secuencia de eventos, entre los objetos que se relacionan durante el flujo de acciones de un escenario. Si los casos de uso tienen varios flujos o secciones, lo más útil es crear un diagrama de secuencia para cada escenario que se encuentre en los casos de uso. [Anexo 6]

3.3.4 Diagramas de Clases Web.

Los diagramas de clases web están orientados a la lógica del negocio y su estado, las reglas del negocio y los requerimientos, lo que brinda la información necesaria para implementar una aplicación web que resuelva las necesidades del negocio.

En la propuesta se elaboró un diagrama de clases web para cada caso de uso del sistema proporcionando una mejor comprensión de las relaciones entre los componentes. [Anexo 7].

En el Anexo 8 se muestran las descripciones de las clases gestoras que se utilizaron en la programación del sistema de Gestión de Solicitudes de Recursos Audiovisuales. [Anexo 8].

3.4 Patrones de Diseño (*Grasp*).

La calidad de diseño de la interacción de los objetos y la asignación de responsabilidades presenta variaciones. Las decisiones poco acertadas dan origen a sistemas y componentes frágiles y difíciles de mantener, entender, reutilizar o extender. Una implementación hábil se funda en los principios cardinales que rigen un buen diseño orientado a objetos. En los patrones GRASP se codifican algunos de ellos, que se aplican al preparar los diagramas de interacción, cuando se asignan las responsabilidades o durante ambas actividades, describen los principios fundamentales de la asignación de responsabilidades a objetos, expresados en forma de patrones. [18]

El diseño de la aplicación debe estar orientado fundamentalmente a los usuarios finales. En la construcción de la aplicación se tuvieron en cuenta diferentes patrones de diseño, a continuación se especifican:

Experto: Se utiliza para asignar responsabilidades al experto en información, la clase que contiene la información necesaria para llevar a cabo la tarea es la experta.

Creador: Plantea que a una determinada clase se le puede asignar la responsabilidad de crear una instancia de otra clase siempre que la misma pueda agregar, contener, registrar y utilizar específicamente objetos de la clase.

3.5 Estándares de Interfaz de Aplicación.

Para el diseño de la interfaz de usuario del sistema de Gestión de Solicitudes de Recursos Audiovisuales se utilizó estilos CSS porque permiten que las páginas tengan menor peso y mayor velocidad de transferencia, además ocupan un menor ancho de banda.

Se trata en todo momento de lograr un contraste entre los colores azul y blanco, y otros colores claros para disminuir el cansancio de la vista del usuario, además proporciona una interfaz amigable.

Los menús se encuentran en el panel izquierdo de la aplicación para garantizar la comodidad al usuario. Para hacer los formularios más pequeños, donde tienden a ser muy grandes solo se muestran algunos datos de las medias, si el usuario desea verlos todos, sólo tiene el que dar clic sobre vínculo “Detalles” el cual aparece al lado de cada media de la solicitud en las pantallas.

3.6 Estándares de Codificación.

Para garantizar la uniformidad del código dentro de la aplicación se estableció el estilo de código que se muestra a continuación.

El nombre de las funciones se escribe con minúscula. Si el nombre de la función es compuesto la primera palabra se mantiene con minúscula y la primera letra de la siguiente palabra con mayúscula. Ejemplo: insertarUsuario().

Las constantes se declaran con mayúsculas. Ejemplo: **define** ('CONSTANTE', 'valor');

Las variables se definen **var** \$nombre;

El nombre de las clases empieza con “T” mayúscula seguido de un sustantivo comenzando con mayúscula. Ejemplo: **class** TUsuario.

Las llaves de las funciones y las condiciones se escriben en una línea aparte para garantizar la legibilidad del código. Ejemplo:

```
if($variable != 0)
```

```
{  
    while(contador > 0)  
    {  
        realizarFuncion();  
    }  
}
```

Los comentarios sencillos se escriben en una línea comenzando por *//*, mientras que para comentar varias líneas se emplean */* */*.

Ejemplo:

```
//-----  
//Comentario de una sola línea  
//-----  
/*Comentario de varias  
Líneas*/
```

Los espacios entre los operadores se establecen de la siguiente manera:

```
if($b == $c)  
{  
    $y = 5;  
    $y = ($b + 1) * $c;  
    llamarFuncion($y);  
}
```

Se define la indentación de la siguiente forma:

```
if($b != 1)  
{
```

```
    echo("b es distinto de 1");  
}
```

3.7 Formatos de Reportes.

Los reportes se representan en forma de tablas, donde los registros se muestran organizados según las opciones que haya marcado el usuario.

3.8 Tratamiento de errores.

El tratamiento de errores es un proceso importante a la hora de implementar un sistema informático porque es la forma de notificar que se produce algún error ante cualquier situación desfavorable que pueda presentarse. Un tratamiento de errores adecuado garantiza el buen funcionamiento de un sistema, complementándose con el grado de la satisfacción de los usuarios al interactuar con él.

En el sistema de gestión de solicitudes de recursos audiovisuales se utiliza la validación en la entrada de datos haciendo uso de JavaScript para evitar que se interpreten datos erróneos innecesariamente.

3.9 Seguridad.

La seguridad en la implementación del sistema de Gestión de Solicitudes de Recursos Audiovisuales formó parte importante de los objetivos propuestos debido a la repercusión que puede tener la no implementación de una seguridad adecuada en el sistema para el uso del equipo de trabajo de Gestión de Medias en la DCAV teniendo presente que los datos que se manipulan en dicho proceso son de vital importancia para el buen funcionamiento de la dirección, garantizando la disponibilidad, confidencialidad e integridad de la información.

La seguridad del sistema está basada en la implementación de sesiones para restringir el acceso no autorizado a las funcionalidades. Las contraseñas de los usuarios se envían encriptadas desde el navegador del cliente hasta el servidor web utilizando el algoritmo MD5 ya que es un algoritmo irreversible y si se produce algún ataque en la red el atacante no podrá obtener la contraseña. Asimismo se almacenan encriptadas en la base de datos, garantizando que sólo el usuario conozca su contraseña.

Periódicamente se realizarán copias de seguridad de la base de datos en el servidor, en las horas de menor actividad, para si se produce algún fallo o ataque se pueda recuperar la última versión de los datos.

3.10 Concepción General de la ayuda.

La implementación de la ayuda en cualquier aplicación web es importante porque es la manera de proporcionarle al usuario una guía o explicación de cómo funciona el software, facilitándoles el manejo de las actividades al interactuar con el sistema.

El Sistema de Gestión de Solicitudes de Recursos Audiovisuales posee una ayuda donde los usuarios podrán hacer uso de ella haciendo clic sobre el vínculo “Ayuda” que se encuentra en el menú superior de la interfaz. Una vez dentro el contenido aparecerá dividido por secciones. En cada sección se explica detalladamente como interactuar con cada pantalla que responde a las funcionalidades del sistema.

3.11 Conclusiones.

En este capítulo se definieron los actores, casos de uso y diagramas de casos de uso del sistema. Se realizaron los diagramas de clases del diseño y los diagramas de secuencia para cada caso de uso. Se identificaron las clases persistentes y se generó el modelo de clases.

También se definió como estilo arquitectónico la arquitectura en tres capas por las ventajas que proporcionan a la hora de implementar un sistema informático. Se analizaron los patrones que se usan para desarrollar la aplicación web para la propuesta de solución.

4

CAPÍTULO Implementación y Prueba

4.1 Introducción.

En el presente capítulo se realizan el modelo de despliegue e implementación. También se realizan las pruebas sobre las interfaces de la aplicación a través de las pruebas de caja negra.

4.2 Modelo de Despliegue.

El modelo de despliegue muestra la disposición física de los distintos nodos que componen un sistema y el reparto de los componentes sobre dichos nodos. Presenta diferentes estereotipos para precisar los equipos, representar las conexiones y poder relacionarlos en un diagrama, por ejemplos dispositivos, procesadores y memorias. El diagrama de despliegue del sistema de Gestión de Solicitudes de Recursos Audiovisuales se puede apreciar en la siguiente imagen.

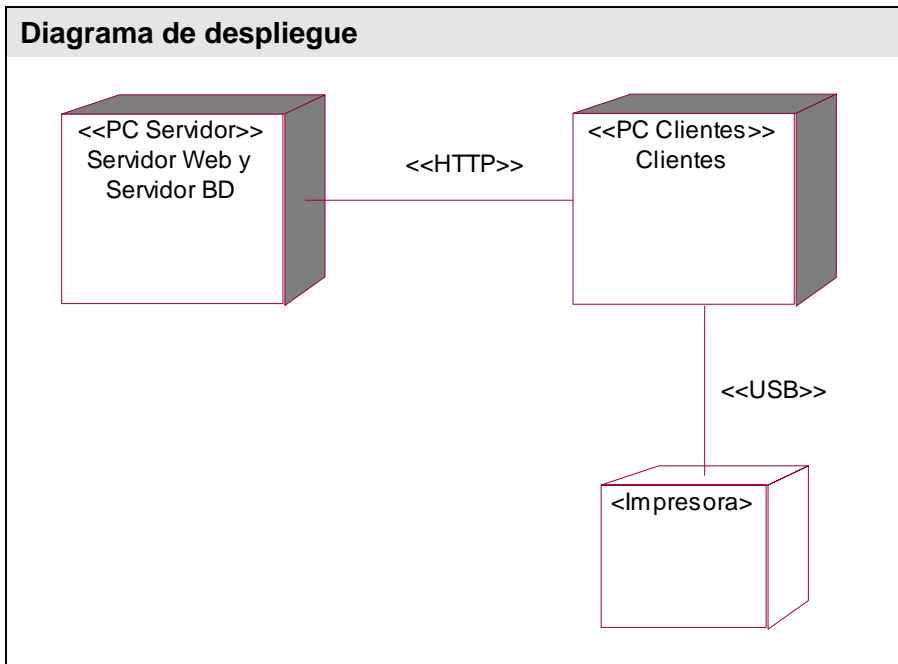


Figura. 9 Diagrama de Despliegue

4.3 Diagramas de Componentes.

Los diagramas de componentes permiten modelar la estructura del software y la dependencia entre componentes. Los componentes pueden corresponder código fuente, binario o ejecutable:

- Ejecutable.
- Biblioteca.
- Tabla.
- Archivo.
- Documento.
- Página Web.

Las relaciones de dependencia indican que un componente utiliza otro, por lo cual depende de él. Estos diagramas se usan principalmente para mostrar la estructura de alto nivel del modelo de implementación, definiendo subsistemas y sus dependencias cuando se requiere importar código y organizar los subsistemas en capas. [Anexo 9].

4.4 Modelo de prueba.

Las pruebas son realizadas a las interfaces o al código fuente de las aplicaciones para disminuir los errores. Existen dos tipos: las pruebas de caja blanca y caja negra. Las pruebas de caja blanca son encargadas de comprobar los caminos lógicos del software, lo que permite determinar si el estado real coinciden con el esperado o mencionado, mientras que las pruebas de caja negra se realizan sobre la interfaz del software con el objetivo de comprobar que las funciones del software son operativas, y que los datos que se introducen son validados correctamente produciendo el resultado correcto.

A continuación se muestran los casos de prueba para los casos de uso “Llenar Solicitud” y “Gestionar Medias”, mostrando primeramente los casos validos y no validos seguidos de los caso de prueba para cada caso de uso. Los textos mostrados en color rojo son los casos no válidos.

Caso de Uso: Llenar Solicitud de Animación.

Condición de Entrada	Casos Válidos	Casos no Válidos
Nombre de la solicitud	Letras y números	No introducir el nombre de la solicitud. Caracteres extraños como ; @ : / *etc.
Tipo de Solicitud	Seleccionar el tipo de solicitud	No seleccionar el tipo de solicitud
Proyecto	Seleccionar el proyecto	No seleccionar el proyecto
Formato	Seleccionar el formato	No seleccionar el formato
Valores	Letras y números	Caracteres extraños como ; @ : / * etc
Id	Letras y números	No introducir el id Caracteres extraños como ; @ : / * etc
Descripción	Cualquier caracter	No introducir la descripción
Origen	Cualquier caracter	No tiene

Caso de uso: Llenar Solicitud
Caso de prueba: Llenar la solicitud de animación entrando correctamente todos los datos.
<p>Entrada: El cliente introduce correctamente los datos necesarios para llenar una solicitud de animación.</p> <p>Nombre de la Solicitud: A Jugar Tipo de solicitud: Animaciones Proyecto: PDVS Formato: swf Valores: 800x600 Id: A1 Descripción: Movimientos de aprobación de la mascota, cara sonriente; en el caso de responder correctamente. Origen: La animación se encuentra en el sitio http://www.cubasi.cu.</p>
Resultado: Se inserta la solicitud en la base de datos.
Condiciones: Los datos para llenar una solicitud de animaciones deben estar en el rango de los valores válidos.

Caso de uso: Llenar Solicitud
Caso de prueba: Llenar la solicitud de animaciones entrando algún dato erróneo.
<p>Entrada: El cliente introduce incorrectamente alguno de los datos necesarios para llenar una solicitud de animación.</p> <p>Nombre de la Solicitud: Art_animación1 @ Tipo de solicitud: "Campo no seleccionado" Proyecto: SIS Formato: fla Valores: "Campo omitido" Id: A2 Descripción: "Campo omitido" Origen:</p>
Resultado: El sistema muestra un mensaje de alerta notificando el error.
Condiciones:

Caso de Uso: Llenar Solicitud Locuciones.

Condición de Entrada	Casos Válidos	Casos no Válidos
Nombre de la solicitud	Letras y números	No introducir el nombre de la solicitud Caracteres extraños como ; @ : / * etc
Tipo de Solicitud	Seleccionar el tipo de solicitud	No seleccionar el tipo de solicitud
Proyecto	Seleccionar el proyecto	No seleccionar el proyecto
Formato	Seleccionar el formato	No seleccionar el formato
Valores	Letras y números	Caracteres extraños como ; @ : / * etc
Id	Letras y números	No introducir el id Caracteres extraños como ; @ : / * etc
Origen	Cualquier caracter	No tiene
Texto	Letras y números	No introducir el texto Caracteres extraños como ; @ : / * etc
Personajes que intervienen	Letras	Números Caracteres extraños como ; @ : / * etc

Caso de uso: Llenar Solicitud
Caso de prueba: Llenar la solicitud de locución entrando correctamente todos los datos.
<p>Entrada: El cliente introduce correctamente los datos necesarios para llenar una solicitud de locución.</p> <p>Nombre de la Solicitud: Multisaber Tipo de solicitud: Locuciones Proyecto: Tesis Formato: wav Valores: 400x300 Id: L3 Origen: El texto de la locución se encuentra en la base de datos DataCenter de las Ciencias Informáticas</p>

<p>Texto: Bienvenidos al modulo 4 “Entretenimientos”, donde podrás realizar sopas de palabras, crucigramas y diferentes juegos que le permitirán ampliar sus conocimientos sobre Teleinformática.</p> <p>Personajes que intervienen: Voz en Off Ricardo Alarcón Quezada.</p>
<p>Resultado: Se inserta la solicitud en la base de datos.</p>
<p>Condiciones: Los datos para llenar una solicitud de locuciones deben estar en el rango de los valores válidos.</p>

<p>Caso de uso: Llenar Solicitud</p>
<p>Caso de prueba: Llenar la solicitud de locuciones entrando algún dato erróneo.</p>
<p>Entrada: El cliente introduce incorrectamente algunos datos necesarios para llenar una solicitud de locución.</p> <p>Nombre de la Solicitud: “Campo omitido”</p> <p>Tipo de solicitud: locuciones</p> <p>Proyecto: “Campo omitido”</p> <p>Formato: waff</p> <p>Valores: “Campo omitido”</p> <p>Id: L4</p> <p>Origen:</p> <p>Texto: Bienvenidos @x130= Yo”, donde podrás realizar sopas de palabras, crucigramas **</p> <p>Personajes que intervienen:</p>
<p>Resultado: El sistema muestra un mensaje de alerta notificando el error.</p>
<p>Condiciones:</p>

Caso de Uso: Llenar Solicitud Digitalizaciones.

Condición de Entrada	Casos Válidos	Casos no Válidos
Nombre de la solicitud	Letras y números	No introducir el nombre de la solicitud Caracteres extraños como ; @ : / *etc
Tipo de Solicitud	Seleccionar el tipo de solicitud	No seleccionar el tipo de solicitud
Proyecto	Seleccionar el proyecto	No seleccionar el proyecto
Formato	Seleccionar el formato	No seleccionar el formato

Valores	Letras y números	Caracteres extraños como ; @ : / * etc
Id	Letras y números	No introducir el id Caracteres extraños como ; @ : / * etc
Nombre Requerido	Letras y números	No introducir el nombre requerido Caracteres extraños como ; @ : / * etc

Caso de uso: Llenar Solicitud
Caso de prueba: Llenar la solicitud de digitalización entrando correctamente todos los datos.
<p>Entrada: El cliente introduce correctamente los datos necesarios para llenar una solicitud de digitalización.</p> <p>Nombre de la Solicitud: Constitución2 Tipo de solicitud: Digitalizaciones Proyecto: SIS Formato: sdf Valores: 400x300 Id: D5 Nombre Requerido: Digitalizaciones_matemáticas</p>
Resultado: Se inserta la solicitud en la base de datos.
Condiciones: Los datos para llenar una solicitud de digitalizaciones deben estar en el rango de los valores válidos.

Caso de uso: Llenar Solicitud
Caso de prueba: Llenar la solicitud de digitalizaciones entrando algún dato erróneo.
<p>Entrada: El cliente introduce incorrectamente algunos datos necesarios para llenar una solicitud de digitalización.</p> <p>Nombre de la Solicitud: "Campo omitido" Tipo de solicitud: digitalizaciones Proyecto: "Campo omitido" Formato: sdf Valores: "Campo omitido" Id: D6</p>

Nombre Requerido: "Campo omitido"
Resultado: El sistema muestra un mensaje de alerta notificando el error.
Condiciones:

Caso de Uso: Llenar Solicitud Sonidos.

Condición de Entrada	Casos Válidos	Casos no Válidos
Nombre de la solicitud	Letras y números	No introducir el nombre de la solicitud Caracteres extraños como ; @ : / *etc
Tipo de Solicitud	Seleccionar el tipo de solicitud	No seleccionar el tipo de solicitud
Proyecto	Seleccionar el proyecto	No seleccionar el proyecto
Formato	Seleccionar el formato	No seleccionar el formato
Resolución	Letras y números	No introducir la resolución
Frecuencia	Letras y números	No introducir la resolución
Tipo según canal		No introducir la resolución
Id	Letras y números	No introducir el id Caracteres extraños como ; @ : / * etc
Descripción	Cualquier caracter	No introducir la descripción
Origen	Cualquier caracter	No tiene
Tipo de sonido	Seleccionar el tipo de sonido	No seleccionar el tipo de sonido
Título	Letras y números	No introducir el título Caracteres extraños como ; @ : / * etc
Autor	Letras	No introducir el autor Números Caracteres extraños como ; @ : / * etc
Categoría	Seleccionar la categoría	No seleccionar la categoría

Caso de uso: Llenar Solicitud
Caso de prueba: Llenar la solicitud de sonido entrando correctamente todos los datos.
<p>Entrada: El cliente introduce correctamente los datos necesarios para llenar una solicitud de sonido.</p> <p>Nombre de la Solicitud: Sonido123 Tipo de solicitud: sonidos Proyecto: Gel Formato: mp3 Resolución: 16 bits Frecuencia: 44KHz Tipo según canal: mono Id: S15 Descripción: Canción infantil. Tipo de sonido: Música Título: Estela el granito de canela Autor: Liuba Maria Hebia Categoría: Infantil</p>
Resultado: Se inserta la solicitud en la base de datos.
Condiciones: Los datos para llenar una solicitud de sonido deben estar en el rango de los valores válidos.

Caso de uso: Llenar Solicitud
Caso de prueba: Llenar la solicitud de sonidos entrando algún dato erróneo.
<p>Entrada: El cliente introduce incorrectamente los datos necesarios para llenar una solicitud de sonido.</p> <p>Nombre de la Solicitud: "Campo omitido" Tipo de solicitud: sonido Proyecto: PDVS Formato: wav Resolución: 15@ Frecuencia: 150kHz Tipo según canal: estéreo Id: S4</p>

Descripción: "Campo omitido" Tipo de sonido: "Campo omitido" Título: Estela el granito de canela Autor: Liuba@yahoo.com Categoría: "Campo omitido"
Resultado: El sistema muestra un mensaje de alerta notificando el error.
Condiciones:

Caso de Uso: Llenar Solicitud Videos.

Condición de Entrada	Casos Válidos	Casos no Válidos
Nombre de la solicitud	Letras y números	No introducir el nombre de la solicitud Caracteres extraños como ; @ : / *etc
Tipo de Solicitud	Seleccionar el tipo de solicitud	No seleccionar el tipo de solicitud
Proyecto	Seleccionar el proyecto	No seleccionar el proyecto
Formato	Seleccionar el formato	No seleccionar el formato
Tamaño		
Id	Letras y números	No introducir el id Caracteres extraños como ; @ : / * etc
Descripción	Cualquier caracter	No introducir la descripción
Origen	Cualquier caracter	No tiene

Caso de uso: Llenar Solicitud
Caso de prueba: Llenar la solicitud de video entrando correctamente todos los datos.
Entrada: El cliente introduce correctamente los datos necesarios para llenar una solicitud de video. Nombre de la Solicitud: Video_Comandante Tipo de solicitud: videos Proyecto: VCS Formato: mpeg

<p>Tamaño: 400x300 Id:V6</p> <p>Descripción: Doblaje y escenificación de la canción "El yerberito" por un niño que entra pregonando con una cesta llena de hierbas medicinales. Debe ir con un sombrero de yarey, vestimenta clara, con una pañoleta de color llamativo en la cintura. Los demás niños y niñas (blancos, mestizos y negros) bien vestidos (6 como mínimo) acuden contentos al pregón para adquirir las hierbas medicinales mientras él canta.</p> <p>Origen: El video se encuentra en el sitio http://granma.co.cu/</p>
<p>Resultado: Se inserta la solicitud en la base de datos.</p>
<p>Condiciones: Los datos para llenar una solicitud de video deben estar en el rango de los valores válidos.</p>

<p>Caso de uso: Llenar Solicitud</p>
<p>Caso de prueba: Llenar la solicitud de videos entrando algún dato erróneo.</p>
<p>Entrada: El cliente introduce incorrectamente los datos necesarios para llenar una solicitud de video.</p> <p>Nombre de la Solicitud: "Campo omitido"</p> <p>Tipo de solicitud: videos</p> <p>Proyecto: "Campo omitido"</p> <p>Formato: avi</p> <p>Tamaño: 800x600</p> <p>Id: V6</p> <p>Descripción: "Campo omitido"</p> <p>Origen: El video se encuentra en el sitio http://granma.co.cu/</p>
<p>Resultado: El sistema muestra un mensaje de alerta notificando el error.</p>
<p>Condiciones:</p>

Caso de Uso: Llenar Solicitud Imágenes.

Condición de Entrada	Casos Válidos	Casos no Válidos
Nombre de la solicitud	Letras y números	No introducir el nombre de la solicitud Caracteres extraños como; @ : / *etc
Tipo de Solicitud	Seleccionar el tipo de solicitud	No seleccionar el tipo de solicitud
Proyecto	Seleccionar el proyecto	No seleccionar el proyecto
Formato	Seleccionar el formato	No seleccionar el formato
Tipo	Seleccionar el tipo de imagen	No seleccionar el tipo de imagen
Tamaño	Letras y números	No introducir el tamaño de la imagen Caracteres extraños como ; @ : / *etc
Resolución	Letras y números	No introducir la resolución de la imagen Caracteres extraños como ; @ : / *etc
Id	Letras y números	No introducir el id Caracteres extraños como ; @ : / * etc
Descripción	Cualquier caracter	No introducir la descripción
Origen	Cualquier caracter	No tiene

Caso de uso: Llenar Solicitud**Caso de prueba:** Llenar la solicitud de imagen entrando correctamente todos los datos.**Entrada:** El cliente introduce correctamente los datos necesarios para llenar una solicitud de imagen.

Nombre de la Solicitud: Imagen_Fidel

Tipo de solicitud: imágenes

Proyecto: VCS

Formato: jpg

Tipo: escaneada

<p>Tamaño: 800x600 Resolución: 72 dpi Id: I12</p> <p>Descripción: La imagen muestra al comandante Fidel Castro junto a los estudiantes de 5to año de las Ciencias Informáticas.</p> <p>Origen: La imagen se encuentra en la página principal del portal de la UCI la dirección es http://intranet/cgi_bin/principal.pl</p>
<p>Resultado: Se inserta la solicitud en la base de datos.</p>
<p>Condiciones: Los datos para llenar una solicitud de imagen deben estar en el rango de los valores válidos.</p>

<p>Caso de uso: Llenar Solicitud</p>
<p>Caso de prueba: Llenar la solicitud de imágenes entrando algún dato erróneo.</p>
<p>Entrada: El cliente introduce incorrectamente los datos necesarios para llenar una solicitud de imagen.</p> <p>Nombre de la Solicitud: Imagen_Fidel Tipo de solicitud: imágenes Proyecto: SIS Formato: png Tipo: original Tamaño: "Campo omitido" Resolución: "Campo omitido" Id: I3 Descripción: "Campo omitido"</p> <p>Origen: La imagen se encuentra en la página principal del portal de la UCI la dirección es http://intranet/cgi_bin/principal.pl</p>
<p>Resultado: El sistema muestra un mensaje de alerta notificando el error.</p>
<p>Condiciones:</p>

Caso de Uso: Validar Redacción Solicitud.

Condición de Entrada	Casos Válidos	Casos no Válidos
Id	Seleccionar el Id	No seleccionar ningún Id
Estado	Seleccionar el estado	No seleccionar ningún estado
Observaciones	Letras y números	Caracteres extraños como ; @ : / *etc
Observaciones de la solicitud	Letras y números	Caracteres extraños como ; @ : / *etc
Fecha de Compromiso	Números	Letras Caracteres extraños como ; @ : / *etc
Responsable	Letras y números	Caracteres extraños como ; @ : / *etc

Caso de uso: Validar Redacción Solicitud.
Caso de prueba: Validar redacción solicitud entrando correctamente todos los datos.
<p>Entrada: El gestor introduce correctamente los datos necesarios para validar la redacción correcta de las medias una solicitud.</p> <p>Id: Seleccionado.</p> <p>Estado: Rechazada.</p> <p>Observaciones: La media contiene errores en la descripción.</p> <p>Observaciones de la Solicitud:</p> <p>Fecha de Compromiso: 15-6-2007</p> <p>Responsable: Lisbell Manresa Nuñez.</p>
Resultado: Se aceptan las medias de la solicitud.
Condiciones: Los datos están en el rango de los valores válidos.

Caso de uso: Validar Redacción Solicitud.
Caso de prueba: Validar redacción solicitud entrando algún dato erróneo.
<p>Entrada: El gestor introduce incorrectamente los datos necesarios para validar la redacción correcta de las medias una solicitud.</p> <p>Id: Seleccionado Estado: "Campo omitido" Observaciones: Observaciones de la Solicitud: Fecha de Compromiso: 15-6-2007 Responsable: "Campo omitido"</p>
Resultado: El sistema muestra un mensaje de alerta notificando el error.
Condiciones:

Caso de Uso: Gestionar Medias.

Condición de Entrada	Casos Válidos	Casos no Válidos
Id	Seleccionar el Id	No seleccionar ningún Id
Estado	Seleccionar el estado	No seleccionar ningún estado
Observaciones	Letras y números	Caracteres extraños como ; @ : / *etc

Caso de uso: Gestionar Medias
Caso de prueba: Gestionar medias entrando correctamente todos los datos.
<p>Entrada: El gestor introduce correctamente los datos necesarios para gestionar las medias una solicitud.</p> <p>Id: Seleccionado Estado: No gestionada Observaciones: La media no se pudo gestionar porque no se encontró en el DataCenter y ni se pudo producir.</p>
Resultado: Se gestionan las medias de la solicitud.
Condiciones: Los datos están en el rango de los valores válidos.

Caso de uso: Gestionar Medias
Caso de prueba: Gestionar medias entrando algún dato erróneo.
Entrada: El gestor introduce incorrectamente los datos necesarios para gestionar las medias una solicitud. Id: Seleccionado Estado: "Campo omitido" Observaciones:
Resultado: El sistema muestra un mensaje de alerta notificando el error.
Condiciones:

Caso de Uso: Validar Medias

Condición de Entrada	Casos Válidos	Casos no Válidos
Id	Seleccionar el Id	No seleccionar ningún Id
Aceptación	Seleccionar algún valor en el campo "Aceptación"	No seleccionar ningún valor
Observaciones	Letras y números	Caracteres extraños como ; @ : / *etc

Caso de uso: Validar Medias
Caso de prueba: Validar medias entrando correctamente todos los datos.
Entrada: El validador introduce correctamente los datos necesarios para validar las medias una solicitud. Id: Seleccionado Aceptación: Si Observaciones:
Resultado: Se termina la solicitud.
Condiciones: Los datos están en el rango de los valores válidos.

Caso de uso: Validar Medias
Caso de prueba: Validar medias entrando algún dato erróneo.
Entrada: El validador introduce incorrectamente los datos necesarios para validar las medias una solicitud. Id: "Campo omitido" Aceptación: Si Observaciones:
Resultado: El sistema muestra un mensaje de alerta notificando el error.
Condiciones:

Debido a que son muchos los datos que hay que validar, se decidió realizar un caso de prueba general para ilustrar algunos de los casos desfavorables que se pueden encontrar.

4.5 Conclusiones.

En este capítulo se obtuvo el modelo de datos del sistema a partir de la definición del diagramas de clases persistentes, quedando definidas todas las tablas de la base de datos con sus atributos y relaciones. Además se especifican algunas concepciones como la concepción de la ayuda y seguridad del sistema, el tratamiento de errores y el formato de los reportes.

También se aplicaron las pruebas de caja negra a diferentes interfaces del sistema para comprobar la funcionalidad del sistema de Gestión de Solicitudes de Recursos Audiovisuales. Se definió la distribución física de los nodos en el diagrama de despliegue de la aplicación y la estructura del software y dependencia entre componentes en los diagramas de componentes.

Conclusiones Generales

En el trabajo desarrollado se dio cumplimiento a los objetivos trazados, implementándose una aplicación web al nivel que se requería, siguiendo paso a paso los requisitos solicitados por el cliente. Se demostró la necesidad del diseño e implementación de un sistema para la gestión del flujo de trabajo **Gestión de Medias** dentro de los tantos servicios que brinda la DCAV en nuestra universidad.

Dicha aplicación brinda un mejor soporte para el desarrollo de todo el proceso que se lleva a cabo a la hora de realizar solicitudes de recursos audiovisuales a la DCAV, proporciona seguridad, y rapidez en el proceso de trabajo con las solicitudes. Además de contener la información en un ambiente agradable a la vista de los usuarios, fácil de comprender y una ayuda online, pues cumple con los principios del diseño propuestos.

A partir del análisis de las tendencias y tecnologías actuales que se utilizan para el desarrollo de aplicaciones informáticas, se elaboró como propuesta la utilización de PHP como lenguaje de programación en conjunto de Smarty como motor de plantilla, además de utilizar MySQL como gestor de base datos y Apache como servidor web para la construcción de la propuesta de solución, en conjunto con otras herramientas y tecnologías y las hojas de estilo CSS para hacer las páginas más ligeras.

Se optó como metodología idónea para llevar a cabo todo el proceso de desarrollo del software a RUP (Proceso Unificado de Rational), utilizando las representaciones de UML para modelar las distintas fases por las que transita el proyecto que se desea desarrollar.

Se realizó la modelación del negocio propuesto, definiendo los actores y trabajadores así como las actividades que son objeto de automatización. Se definieron los requisitos funcionales y no funcionales del sistema para la posterior estructuración de los casos de uso del sistema.

Se realizó el análisis y diseño del sistema a través de diagramas de clases web, clases persistentes, entre otros. Se generó el modelo de datos a partir del diagrama de clases persistentes, el cual muestra la representación física de la base de datos del sistema.

Además se plantearon los principios de diseño a través de los estándares de interfaz de aplicación, de codificación, así como los patrones de diseño y tratamientos de errores. Se construyó finalmente el diagrama de despliegue y componentes, y se realizaron las pruebas de caja negra a las diferentes interfaces del sistema.

Luego de este largo proceso de trabajo se puede decir que SISRA es un sistema que brinda la solución requerida ante la problemática que la originó y que su utilización representará una mejora considerable en el control y eficiencia de los procesos que automatiza dentro del flujo de trabajo de **Gestión de Medias**.

Recomendaciones

Se recomienda hacer un estudio más profundo sobre el proceso de Gestión de Recursos Audiovisuales de la DCAV para analizar la posible automatización de otras funciones que no se implementaron en la propuesta de solución.

Analizar la posibilidad de integrar SISRA al sistema GIA (Gestión de Información Audiovisual) perteneciente a la DCAV.

También se recomienda implementar otras funcionalidades a la aplicación, tales como:

- Dar la posibilidad de listar todos los usuarios que pertenecen a un determinado rol.
- Realizar reportes donde se muestren más salidas.
- Implementar un módulo de análisis estadístico, sobre el procesamiento de solicitudes de recursos audiovisuales.
- Permitir conocer cuales son las solicitudes que se encuentran en el historial.
- Paginar los resultados obtenidos de la base de datos.

Se recomienda buscar otros mecanismos de aviso además del uso del correo electrónico.

Referencias Bibliográficas

- [1] García, Eugenio. *Workflow y UML* [en línea]. 1999. [Consultado el: 19/12/2006]. Disponible en: <http://www.willydev.net/descargas/articulos/general/WorkFlowUML.pdf>
- [2] González, Zorelly. Pichardo, Karen. *Gerencia del Conocimiento y las nuevas tendencias en Sistemas de Control y Gestión*. [en línea]. Instituto de Ingeniería. [Consultado el:12/03/2007]. Disponible en: <http://www.fii.org/jornadas/Trabajos%20Jornadas%20Tecnicas/Cies/JT-E-05.doc>
- [3] Piñeiro Yadenis. Martínez, Yaillet. Lorente R, Abel E. Báez D, Delmis C. *Sistema Metodológico para el desarrollo de Software.2006 Educativo* [Consultado el: 15/11/2006]. 67 p.
- [4] Calero S, Manuel. [en línea]. *Una explicación de la programación extrema (XP)* 2003. [Consultado el: 19/01/2007]. Disponible en: <http://www.willydev.net/descargas/prev/ExplicaXP.pdf>. 18 p.
- [5] Molpeceres, Alberto. *Procesos de desarrollo: RUP, XP, FDD* [en línea]. 2003. [Consultado el: 19/01/2007]. Disponible en: <http://www.javahispano.org/articles.print.action?id=76> .11p.
- [6] Colectivo de Autores. *Tecnología ASP.NET in Conferencia 8* [en línea]. 2006-2007. Universidad de las Ciencias Informática. [Consultado el: 13/02/2007].
- [7] Alarcón, José M. *¿Qué es esto del "code-beside" que trae ASP.NET 2.0?* [en línea]. 2005. [Consultado el: 20/01/2007]. Disponible en: http://www.netveloper.com/contenido2.aspx?IDC=209_0
- [8] Cortes C, Carlos *¿Porque PHP se esta comiendo al ASP?* [en línea]. 2002. [Consultado el: 20/01/2007]. Disponible en: <http://bulma.net/body.phtml?nIdNoticia=1363>
- [9] Colectivo de Autores. *Personalización de Aplicaciones in Conferencia 13*. 2006-2007. Universidad de las Ciencias Informática. [Consultado el: 13/02/2007].
- [10] Ohrt, Monte. Zmievski, Andrei. *Smarty Manual* [en línea]. New Digital Group, 2005. [Consultado el: 26/01/2007]. Disponible en: <http://www.smarty.net> .185 p.
- [11] Colectivo de Autores. *Introducción al SQL in Conferencia 6* [en línea]. 2006-2007. Universidad de las Ciencias Informática. [Consultado el: 13/02/2007]. Disponible en: http://teleformacion.uci.cu/file.php/45/CLASES/Conferencias/Conferencia_6/C6-Introduccion_al_SQL.Sentencias_de_definicion_DDL.pdf
- [12] Herrarte S, Pedro. *Componentes del lenguaje SQL* [en línea]. 2005. [Consultado el: 20/01/2007]. Disponible en: http://www.devjoker.com/asp/ver_contenidos.aspx?co_contenido=10
- [13] Pérez, Jose M. *¿Qué es MySQL?* [en línea]. 2005. [Consultado el: 20/01/2007]. Disponible en: <http://www.espestudio.com/articulo/desarrollo-web/bases-de-datos-mysql/Que-es-MySQL.htm>

- [14] Marrero, Eduardo. *¿Qué es MySQL?* [Consultado el: 20/01/2007]. Disponible en: <http://www.mysql-hispano.org/page.php?id=2>.
- [15] *Programación por capas* [en línea] [Consultado el: 30/01 de 2007]. Disponible en: http://es.wikipedia.org/wiki/Programaci%C3%B3n_por_capas
- [16] Ronda A, Yoelys. Cobas S, Kadir. Marrero V, Daymel. *El modelo cliente-servidor de tres capas para el trabajo en redes de información*. [en línea] 2002. [Consultado el: 15/02/2007]. Disponible en: <http://espejos.unesco.org.uy/simplac2002/Ponencias/ambientes%20digitales/AD067.doc>
- [17] *Metodologías y Herramientas de desarrollo. Rational Rose. Informe del producto* [en línea] [Consultado el: 21/01/2007]. Disponible en: http://www.indudata.com/1rational_rose.htm
- [18] Craig, Larman. *UML y Patrones. Introducción al análisis y diseño orientados a objetos*. Primera edición en español. México. Prentice Hall, 1999. 499 p.

Bibliografía

1. Hernández L, Rolando A. Cuello H, Sayda. *EL Paradigma Cuantitativo de la Investigación Científica*. Ciudad de la Habana, Editorial Universitaria, 2002. 114 p.
2. Pressman, Roger S. *Ingeniería de Software, un enfoque práctico*. Quinta edición. McGraw-Hill, 2002.
3. Craig, Larman. *UML y Patrones. Introducción al análisis y diseño orientados a objetos*. Primera edición en español. México. Prentice Hall, 1999. 499 p.
4. Craig, Larman. *UML y Patrones. Una introducción al análisis orientado a objetos*. Segunda edición en español. Prentice Hall. 520 p.
5. Gibbs, Matt. *Almacenador de estado de páginas de ASP.NET 2.0* [en línea]. 2006. [Disponible en: <http://www.microsoft.com/spanish/msdn/articulos/archivo/270306/voices/PgStPrst.aspx>]
6. Janssens, Johan. *Plantillas en Joomla! 1.5. Parte 1* [en línea]. 2006. [Disponible en: http://ayuda.joomlaspanish.org/index.php?option=com_content&task=view&id=425&Itemid=91]
7. Colectivo de Autores. *Clases de Ingeniería de Software 1 y 2* [en línea]. 2005-2006. Universidad de las Ciencias Informáticas. Disponible en: <http://teleformacion.uci.cu/>
8. Trejo M, Janhil A. *Bases de datos* [en línea]. Disponible en: <http://www.monografias.com/trabajos11/basda/basda.shtml>
9. *Migrar una base de datos a MySQL* [en línea]. 2004. Disponible en: <http://www.desarrolloweb.com/articulos/1231.php>

Glosario de Términos

A continuación se muestra el significado de algunos términos que son usados en el trabajo y que pueden ser de difícil comprensión.

- **Medias y recursos audiovisuales:** Recursos que son gestionados en La Dirección de Comunicación Audiovisual durante el flujo de trabajo de Gestión de Recursos Audiovisuales. Las medias pueden ser animaciones, locuciones, digitalizaciones, imágenes, videos y sonidos.
- **Componente:** Es una parte modular de un sistema, desplegable y reemplazable que encapsula implementación y un conjunto de interfaces y proporciona la realización de los mismos. Un componente típicamente contiene clases y puede ser implementado por uno o más artefactos (ficheros ejecutables, binarios, etc).
- **UserStories:** Son una especie de historias que realiza el cliente, las cuales describen escenario del desarrollo del software (modelos).
- **HTML WYSIWYG:** Editor HTML que permite crear diseños de elementos con estilos sin necesidad de saber qué etiqueta corresponde a cada elemento.
- **HiddenFieldPageState:** Almacenador de estados de páginas incluidas dentro de ASP.NET.
- **SessionPageStatePersiste:** Utilizada en ASP.NET, donde se emplean sesiones de servidor asociadas con la sesión del explorador para almacenar los datos.
- **GRASP:** Patrones Generales de Asignación de Responsabilidades Software que guían a los desarrolladores a encontrar los patrones de diseño (Experto, Creador, Bajo Acoplamiento, Alta Cohesión).
- **HTTP:** Es el protocolo de transferencia de hipertexto usado en cada transacción de la Web. El hipertexto es el contenido de las páginas web, y el protocolo de transferencia es el sistema mediante el cual se envían las peticiones de acceso a una página y la respuesta con el contenido. También sirve el protocolo para enviar información adicional en ambos sentidos, como formularios con campos de texto. HTTP es un protocolo sin estado, es decir, que no guarda ninguna información sobre conexiones anteriores.